A STUDY OF SUPERSONIC FLOW PAST VIBRATING SHELLS AND CASCADES

Christian William Brix

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A STUDY OF SUPERSONIC FLOW
PAST VIBRATING SHELLS AND CASCADES

by

Christian William Brix, Jr.

Thesis Advisor: M. F. Platzer

June 1973

A Study of Supersonic Flow Past Vibrating Shells and Cascades

by

Christian William Brix, Jr.
Lieutenant, United States Navy
M.S., Naval Postgraduate School, 1972

Submitted in partial fulfillment of the
requirements for the degree of

AERONAUTICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
June 1973

ABSTRACT

Supersonic flow past oscillating cylindrical shells and oscillating flat-plate cascades is analyzed using linearized characteristics methods. Pressure distributions and generalized aerodynamic forces are computed for arbitrary radius to length ratios, axial and circumferential wave numbers, Mach numbers, and reduced frequencies. Excellent agreement is obtained with a previous solution of the complete unsteady linearized potential equation by Dowell and Widnall using Laplace transform techniques. Also, pressure distributions are computed for flat-plate cascades with subsonic leading edge locus for arbitrary solidity, stagger angle, frequency, and interblade phase angle. For comparison, a two-blade solution is developed using the method of singularities. Good agreement is obtained for this special case.

TABLE OF CONTENTS

Page

LIST OF ILLUSTRATIONS

LIST OF SYMBOLS

A_1	See Equation (G-3)
C	Teipel complex velocity of sound perturbation amplitude.
c	Velocity of sound perturbation
$\bar{c}$	Local velocity of sound
c_∞	Free-stream velocity of sound
$\hat{c}$	Blade chord
C_p	Coefficient of pressure
$\bar{C}_p$	Pressure coefficient amplitude
d	Interblade distance
e	2.718282
$H_n^{(2)}$	Hankel function of the second kind, order n
h	Radial deflection (shell) Blade chord line (cascade)
i	The square root of minus one
J_n	Bessel function of the first kind, order n
k	Reduced frequency
L	Cylinder length
M	Free-stream Mach number
m	Axial mode number
N	Number of blades
n	Circumferential mode number
P	Non-dimensional complex pressure perturbation amplitude
p	Pressure
p_∞	Free-stream pressure

R	Radius to length ratio
r	Radial coordinate
S	Entropy
Δs	Increment of distance along a characteristic
T	Temperature
t	Time
U	Teipel complex streamwise velocity perturbation amplitude
u	Streamwise velocity perturbation
$\bar{u}$	Local streamwise velocity
u_∞	Free-stream velocity
V	Teipel complex normal velocity perturbation amplitude
v	Normal velocity perturbation
W	Velocity in the z-direction
w	z-direction velocity perturbation
x,y,z	Rectangular coordinates
x,r,θ	Cylindrical coordinates
x_0	Elastic axis position
Y_n	Bessel function of the second kind, order n
$Z(x)$	Radial deflection amplitude
α	Mach angle
β	Stagger angle
δ	Interblade phase angle
ξ,η	Characteristic coordinates
ν	Grid fineness ratio
ρ	Local density
ρ_∞	Free-stream density

Φ	Velocity perturbation amplitude
ϕ	Velocity potential amplitude
$\psi_m(x)$	Stream-wise mode shape
ψ_1	See Equation (G-3)
ω	Circular frequency
π	3.141593

ACKNOWLEDGMENT

This thesis was written with the close support and continual assistance of Associate Professor M. F. Platzer.

I. INTRODUCTION

Supersonic flutter of skin panels first occurred on the German V-2 missile and has become of increasing engineering importance in recent years with the development of supersonic flight vehicles. Panel flutter, for example, was found to occur on the X-15 Rocket Aircraft and Saturn Space Launch Vehicle. An integral part of the flutter analysis of such panels is the determination of the oscillatory pressure distribution. A number of approaches have been formulated which resulted mostly in approximation theories. A detailed review of the state of the art can be found in Reference 6. Only one "exact" approach was worked out by Dowell and Widnall using the complete linearized unsteady potential equation (Ref. 2). However, this approach requires rather complicated and time-consuming complex integration techniques.

In the present work a method of characteristics procedure is developed to determine supersonic flow past vibrating panels and cylindrical shells resulting in a rapid computation of the pressure distribution and generalized aerodynamic forces.

A second flutter problem has also become of considerable interest very recently. Due to the continuing trend toward higher tip speeds in modern jet engine fans and compressors, the possibility of supersonic blade flutter has become of considerable concern. A recent review of this problem area

has been given in "A Workshop on Aeroelasticity in Turbomachines" (Ref. 13).

In the present study the method of characteristics is applied to compute supersonic flow past a finite flat plate cascade with subsonic leading edge locus. In the past, cascade studies have proved to be a useful tool for the understanding of the complicated three-dimensional, incompressible or subsonic flow through an actual compressor stage. Although it is doubtful that the much more complex mixed transonic-supersonic flow can be modeled by a two-dimensional cascade idealization, the analysis of an infinitely thin flat plate cascade may point the way toward a fuller understanding. The method of characteristics again proved to be a convenient tool for the analysis of supersonic flow past oscillating cascades.

II. PROBLEM FORMULATION

Inviscid adiabatic flow of a gas along the positive x-axis is considered which is perturbed by either A) an airfoil cascade executing small pitch oscillations (Fig. 1) or B) small amplitude vibrations of the surface of a circular cylinder whose axis is aligned with the free-stream direction (Fig. 2).

The following basic equations hold:

The equation of state for a perfect gas,

$$p = \rho RT \tag{1}$$

the conservation equation for mass,

$$\frac{D\rho}{Dt} + \rho\left(\frac{\partial \bar{u}}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) = 0 \tag{2}$$

the conservation of momentum

$$\frac{D\bar{u}}{Dt} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0$$

$$\frac{Dv}{Dt} + \frac{1}{\rho}\frac{\partial p}{\partial y} = 0$$

$$\frac{Dw}{Dt} + \frac{1}{\rho}\frac{\partial p}{\partial z} = 0$$

and the conservation of energy

$$\frac{Ds}{Dt} = 0$$

where $\frac{D}{Dt}$ is the substantial derivative and s is the entropy of the fluid particle.

The assumption of small amplitude vibrations makes it possible to introduce the small perturbation concept.

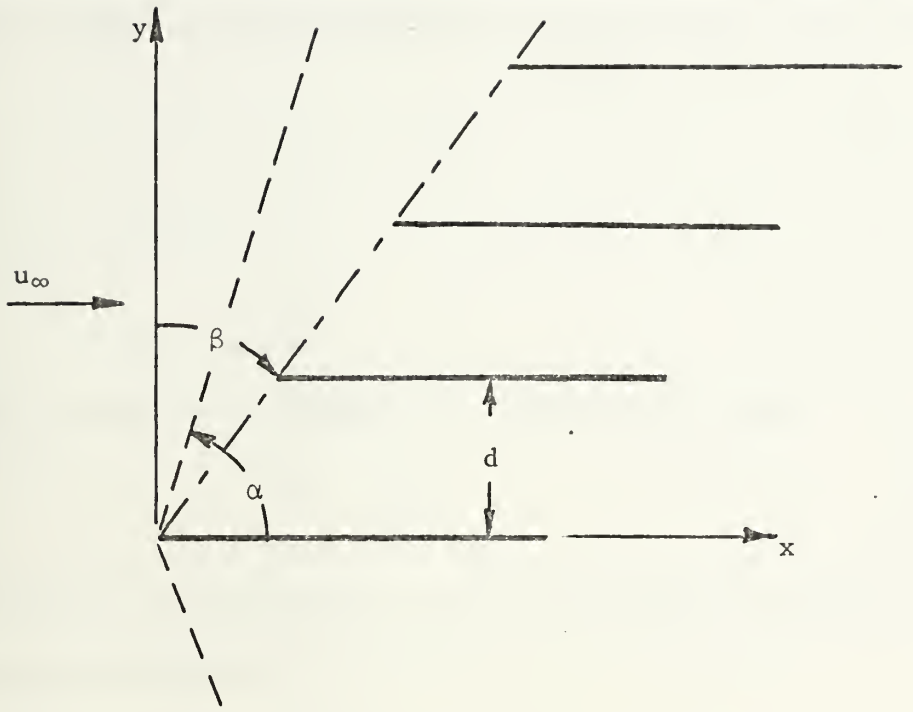


Figure 1. Cascade Configuration

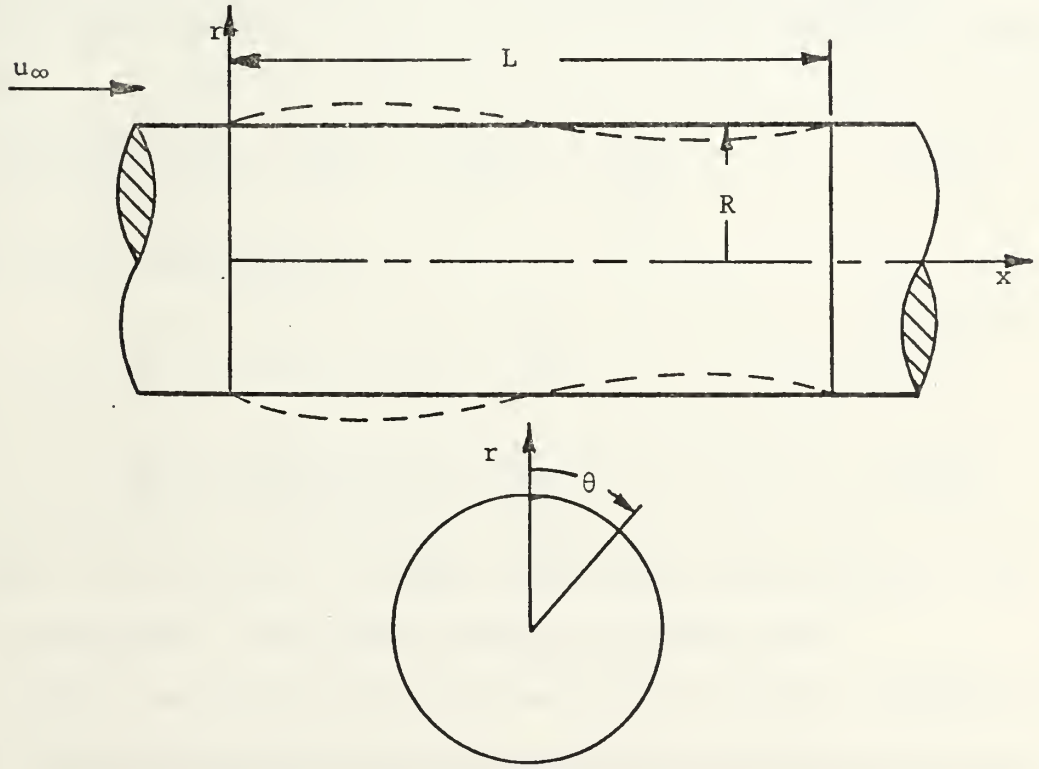


Figure 2. Cylinder Geometry

Therefore, all flow quantities are assumed as small perturbations linearly superimposed on free-stream quantities, i.e.,

$$\bar{u} = u_\infty + u \tag{5}$$

$$V = v \tag{6}$$

$$W = w \tag{7}$$

$$\bar{c} = c_\infty + c \tag{8}$$

and the pressure and density perturbations are:

$$\Delta p = p - p_\infty \tag{9}$$

$$\Delta \rho = \rho - \rho_\infty \tag{10}$$

The pressure perturbation can be expressed by the velocity of sound perturbation, i.e.,

$$p - p_\infty = \frac{2}{\gamma - 1} \rho_\infty c_\infty (\bar{c} - c_\infty) \tag{11}$$

Specializing to planar flow, one has for the continuity equation:

$$\frac{2}{\gamma-1} \frac{\partial c}{\partial t} + \frac{2}{\gamma-1} u_\infty \frac{\partial c}{\partial x} + c_\infty \frac{\partial u}{\partial x} + c_\infty \frac{\partial v}{\partial y} = 0 \tag{12}$$

and the Euler equations:

$$\frac{\partial u}{\partial t} + u_\infty \frac{\partial u}{\partial x} + \frac{2}{\gamma-1} c_\infty \frac{\partial c}{\partial x} = 0 \tag{13a}$$

$$\frac{\partial v}{\partial t} + u_\infty \frac{\partial v}{\partial x} + \frac{2}{\gamma-1} c_\infty \frac{\partial c}{\partial y} = 0 \tag{13b}$$

Equations (12) and (13) are the governing equations for the case of supersonic flow past oscillating cascades.

For the case of supersonic flow past vibrating cylindrical shells, the equivalent single equation for the perturbation

potential, $\Phi(x,r,\theta,t)$ in cylindrical coordinates was taken as the more convenient basic governing equation, i.e.,

$$(1-M^2)\Phi_{xx}+\Phi_{rr}+\frac{1}{r}\Phi_r+\frac{1}{r^2}\Phi_{\theta\theta}-2\frac{M}{c_\infty}\Phi_{xt}-\frac{1}{c_\infty^2}\Phi_{tt} = 0 \qquad (14)$$

For a derivation of this equation reference is made to one of the standard texts, e.g., Bisplinghof, Ashley, and Halfman, "Aeroelasticity", (Ref. 1).

For a complete formulation, proper initial and boundary conditions must be prescribed. The most important ones are A) the flow on the body surface must be tangent at any instant in time and B) Sommerfeld's radiation condition must be satisfied, i.e., waves must propagate away from sources of disturbance.

After properly linearizing the general flow tangency condition, (see Ref. 1) one obtains for the flow tangency condition on a thin airfoil, performing a time dependent small amplitude motion about the y=0 plane,

$$v = \frac{\partial h}{\partial t}+u_\infty\frac{\partial h}{\partial x} \qquad \text{at } y=0 \qquad (15)$$

where $h(x,t)$ is the equation of the airfoil chord line.

For the vibrating cylindrical shell the flow tangency condition over the flexible portion becomes

$$\Phi_r\Big|_{r=R} = \frac{\partial h}{\partial t}+u_\infty\frac{\partial h}{\partial x} \qquad 0\leq x\leq 1$$

$$\Phi_r\Big|_{r=R} = 0 \qquad x<0 \qquad (16)$$

where $h(x,\theta,t)$ in this case describes the radial shell deflection.

With the assumption of a perturbation potential, the momentum equations, equations (3), may be integrated to

$$\frac{\partial \Phi}{\partial t}+\frac{u_\infty^2}{2}+u_\infty\frac{\partial \Phi}{\partial x}+\int\frac{dp}{\rho}=c(t) \tag{17}$$

assuming uniform parallel flow far upstream,

$$c(t)=\frac{u_\infty^2}{2} \tag{18}$$

Also, the density-pressure relationship can be expanded as

$$\rho(p)=\rho_\infty+\left(\frac{d\rho}{dp}\right)(p-p_\infty)+\frac{1}{2}\left(\frac{d^2\rho}{dp^2}\right)=(p-p_\infty)^2+\ldots \tag{19}$$

therefore,

$$\int_{p_\infty}^{p}\frac{dp}{\rho(p)}=\frac{1}{\rho_\infty}\int_{p_\infty}^{p}\left[1-\frac{p-p_\infty}{p_\infty\gamma}\right]dp\cong\frac{p-p_\infty}{\rho_\infty} \tag{20}$$

Hence, the perturbation pressure can be expressed in terms of the perturbation potential, i.e.,

$$p-p_\infty=-\rho_\infty\left[\frac{\partial \Phi}{\partial t}+u_\infty\frac{\partial \Phi}{\partial x}\right] \tag{21}$$

In pressure coefficient form this becomes

$$C_p=\frac{p-p_\infty}{\frac{1}{2}\rho_\infty u_\infty^2}=-\frac{2}{u_\infty^2}\Phi_t-\frac{2}{u_\infty}\Phi_x \tag{22}$$

For further analysis it is convenient to use non-dimensional quantities. Introducing for the oscillating cascade case the blade chord as the unit of length one has the following non-dimensional coordinates:

$$\bar{x} = \frac{x}{\hat{c}}, \quad \bar{y} = \frac{y}{\hat{c}}, \quad \bar{t} = \frac{t\, u_\infty}{\hat{c}} \tag{23}$$

Furthermore, introducing the assumption of harmonic time-dependence, one can define a reduced frequency

$$k = \frac{\omega \hat{c}}{u_\infty} \tag{24}$$

where ω is the circular frequency of blade oscillation.

Following previous work by Teipel (Ref. 3) for single blade oscillations the non-dimensional complex amplitude functions

$$U(\bar{x},\bar{y})e^{i\omega t} = \frac{u}{u_\infty} \tag{25}$$

$$V(\bar{x},\bar{y})e^{i\omega t} = \frac{1}{\sqrt{M^2-1}} \frac{v}{u_\infty} \tag{26}$$

$$C(\bar{x},\bar{y})e^{i\omega t} = \frac{2}{\gamma-1} \frac{1}{M^2} \frac{c}{c_\infty} \tag{27}$$

are introduced. Substituting these equations into the continuity and Euler equations (12) and (13) gives

$$\frac{\partial U}{\partial x} + \sqrt{M^2-1} \frac{\partial V}{\partial y} + M^2 \frac{\partial C}{\partial x} + ikM^2C = 0 \tag{28}$$

$$\frac{\partial U}{\partial x} + \frac{\partial C}{\partial x} + ikU = 0 \tag{29}$$

$$\frac{\partial V}{\partial x} + \frac{1}{\sqrt{M^2-1}} \frac{\partial C}{\partial y} + ikV = 0 \tag{30}$$

where the overbar on x and y is being dropped for convenience in the subsequent analysis.

Introducing similar non-dimensional variables with distances referred to flexible cylinder length, velocities to free-stream speed and time to cylinder length divided by free-stream speed, one obtains the equation for the non-dimensional perturbation potential, i.e.,

$$(1-M^2)\Phi_{xx} + \Phi_{rr} + \frac{1}{r^2}\Phi_{\theta\theta} - 2M^2\Phi_{xt} - M^2\Phi_{tt} = 0 \qquad (31)$$

subject to the boundary condition

$$\Phi_r\Big|_{r=R} = \frac{\partial h}{\partial t} + \frac{\partial h}{\partial x} \qquad 0 \leq x \leq 1$$
$$= 0 \qquad x<0 \qquad (32)$$

If the radial shell deflection is assumed as

$$h(x,\theta,t) = Z(x)\cos n\theta \cdot e^{ikt} \qquad (33)$$

where k is now referred to cylinder length, i.e.,

$$k = \frac{\omega L}{u_\infty} \qquad (34)$$

Then the velocity potential must also have the form,

$$\Phi(x,r,\theta,t) = \phi(x,r)\cos n\theta \cdot e^{ikt} \qquad (35)$$

Substitution of equation (35) into equation (31) then gives

$$(1-M^2)\phi_{xx} + \phi_{rr} + \frac{1}{r}\phi_r - \frac{n^2}{r^2}\phi - 2ikM^2\phi_x + k^2M^2\phi = 0 \qquad (36)$$

subject to

$$\phi_r\Big|_{r=R} = \frac{\partial z}{\partial x} + ikZ \qquad 0 \leq x \leq 1$$
$$= 0 \qquad x<0 \qquad (37)$$

The pressure coefficient in terms of the non-dimensional velocity potential is

$$C_p = -2[ik\Phi + \Phi_x] \qquad (38)$$

and the pressure coefficient amplitude is

$$\bar{C}p = -2[ik\phi + \phi_x] \qquad (38a)$$

These are the basic equations used in the subsequent cylindrical shell vibration analysis.

III. THE METHOD OF CHARACTERISTICS

A. CYLINDRICAL SHELL

In Section II, equation (36) was developed as the partial differential equation for $\phi(x,r)$ that requires solution subject to the boundary condition, equation (37). The following developments are based on a previous paper by Platzer, Brix, and Webster, (Ref. 7).

If M>1, equation (30) is hyperbolic and has real characteristics, which satisfy the ordinary differential equation

$$(M^2-1)dr^2-dx^2 = 0 \tag{39}$$

An arbitrary function F(x,r) has the following derivatives along the two characteristics:

$$\frac{dF}{ds} = F_x \cdot \frac{dx}{ds} + F_r \frac{dr}{ds} = \frac{\sqrt{M^2-1}}{M} F_x \pm \frac{1}{M} F_r \tag{40}$$

where ds is the differential arc-length along the characteristic $ds^2 = M^2dr^2$ and

$$\frac{dx}{ds} = \frac{\sqrt{M^2-1}}{M} \tag{41}$$

and

$$\frac{dr}{ds} = \pm \frac{1}{M} \tag{42}$$

Letting s_1 be the arc-length of the left-running Mach line and s_2 the arc -length of the right-running Mach line, one obtains for the first derivatives,

$$F_1 = \frac{\sqrt{M^2-1}}{M} F_x + \frac{1}{M} F_r \tag{43}$$

and

$$F_2 = \frac{\sqrt{M^2-1}}{M} F_x - \frac{1}{M} F_r \qquad (44)$$

and for the second derivatives

$$-F_{12} = \frac{dF_1}{ds_2} = \frac{1}{M^2} [(1-M^2)F_{xx} + F_{rr}] = -F_{21} \qquad (45)$$

Therefore, equation (30) can be written as

$$\frac{d\phi_1}{ds_2} = \frac{\phi_1-\phi_2}{2rM} - \frac{ikM}{\sqrt{M^2-1}} (\phi_1 + \phi_2) + (k^2 - \frac{n^2}{r^2M^2})\phi \qquad (46)$$

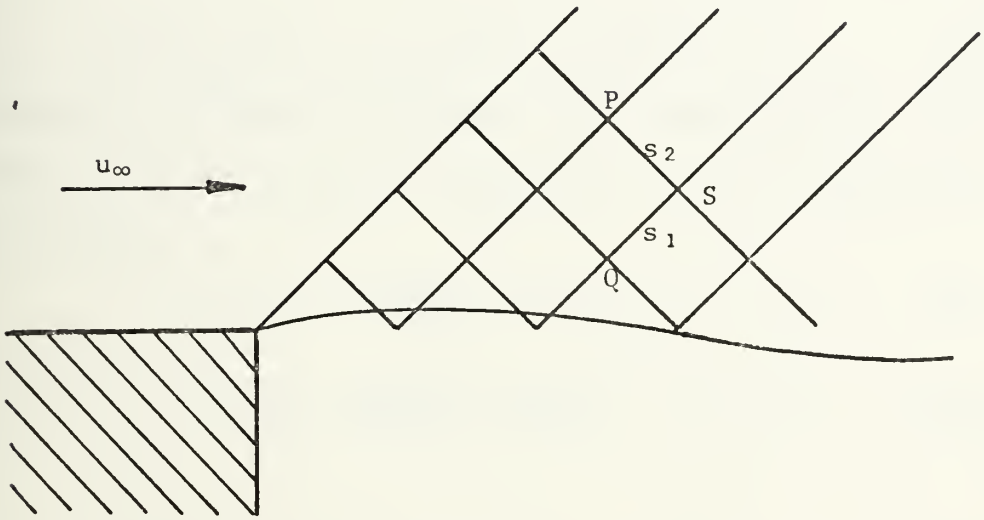


Figure 3. Cylindrical Shell Characteristics Network

The finite difference form of this equation for the points P and S (see Figure 3) on the right-running Mach line is

$$\frac{\phi_1(S) - \phi_1(P)}{\Delta s} = \frac{\phi_1-\phi_2}{2rM} - \frac{ikM}{\sqrt{M^2-1}} (\phi_1+\phi_2) + (k^2-\frac{n^2}{r^2M^2})\phi \qquad (47)$$

Similarly, for the points Q and S on the left-running Mach line one obtains

$$\frac{\phi_2(S) - \phi_2(Q)}{\Delta s} = \frac{\phi_1-\phi_2}{2rM} - \frac{ikM}{\sqrt{M^2-1}} (\phi_1+\phi_2) + (k^2-\frac{n^2}{r^2M^2})\phi \qquad (48)$$

Now solving for the values of ϕ_1 and ϕ_2 at S from the known values at P and Q (Figure 3) the entire flow field may be computed point by point. For this purpose the values for ϕ_1 and ϕ_2 in equation (47) are averaged between P and S, i.e.,

$$\phi_1 = \frac{\phi_1(P) + \phi_1(S)}{2} \tag{49}$$

$$\phi_2 = \frac{\phi_2(P) + \phi_2(S)}{2} \tag{50}$$

and

$$\phi(S) = \phi(P) + \tfrac{1}{2}[\phi_2(P) + \phi_2(S)]\Delta s \tag{51}$$

is obtained by integration along the right-running Mach line.

Similarly, ϕ_1 and ϕ_2 in equation (48) are averaged between Q and S

$$\phi_1 = \frac{\phi_1(Q) + \phi_1(S)}{2} \tag{52}$$

$$\phi_2 = \frac{\phi_2(Q) + \phi_2(S)}{2} \tag{53}$$

and

$$\phi(S) = \phi(Q) + \frac{1}{2}[\phi_1(Q) + \phi_1(S)]\,\Delta s \tag{54}$$

This results in a system of two equations for $\phi_1(S)$ and $\phi_2(S)$, i.e.,

$$A\phi_1(S) + B\phi_2(S) = C \tag{55}$$

$$D\phi_1(S) + E\phi_2(S) = F \tag{56}$$

where

$$A = 1-\frac{\Delta s}{4Mr(S)} + \frac{ikM\Delta s}{2\sqrt{M^2-1}} \tag{57}$$

$$B = \frac{\Delta s}{4Mr(S)} + \frac{ikM\Delta s}{2\sqrt{M^2-1}} - \frac{\Delta s^2}{4}(k^2 - \frac{n^2}{M^2r^2(S)}) \tag{58}$$

$$C = \phi_1(P)[1 + \frac{\Delta s}{4Mr(P)} - \frac{ikM\Delta s}{2\sqrt{M^2-1}}]$$

$$+ \phi_2(P)[- \frac{\Delta s}{4Mr(P)} - \frac{ikM\Delta s}{2\sqrt{M^2-1}} + \frac{\Delta s^2}{4}(k^2 - \frac{n^2}{M^2r^2(S)})] \tag{59}$$

$$D = \frac{\Delta s}{4Mr(S)} - \frac{ikM\Delta s}{2\sqrt{M^2-1}} + \frac{\Delta s^2}{4}(k^2 - \frac{n^2}{M^2r^2(S)})] \tag{60}$$

$$E = - 1 - \frac{\Delta s}{4Mr(S)} - \frac{ikM\Delta s}{2\sqrt{M^2-1}} \tag{61}$$

$$F = \phi_2(Q)[-1 + \frac{\Delta s}{4Mr(Q)} + \frac{ikM\Delta s}{2\sqrt{M^2-1}}]$$

$$+ \phi_1(Q)[- \frac{\Delta s}{4r(Q)M} + \frac{ikM\Delta s}{2\sqrt{M^2-1}} - \frac{\Delta s^2}{4}(k^2 - \frac{n^2}{M^2r^2(S)})]$$

$$+ \phi(Q)[- \frac{\Delta s}{2}(k^2 - \frac{n^2}{r^2(S)M^2}) - \frac{\Delta s}{2}(k^2 - \frac{n^2}{M^2r^2(Q)})] \tag{62}$$

The above procedure will suffice for the computation of ϕ and its derivatives at a general flow field point, but needs to be modified to obtain the values on the initial Mach line and along the cylinder surface.

On the initial Mach line (Figure 3) it is known that $\phi=\phi_1=0$. Hence, equation (56) becomes

$$E\phi_2(S) = F \tag{63}$$

where E is known from equation (61) and F simplifies to

$$F = \phi_2(Q) \left[-1 + \frac{\Delta s}{4Mr(Q)} + \frac{ikM\Delta s}{2\sqrt{M^2-1}}\right] \tag{64}$$

On the cylinder surface the normal velocity is prescribed, i.e.,

$$\phi_r = \frac{M}{2} (\phi_1-\phi_2) = \frac{\partial Z}{\partial x} + ikZ \tag{65}$$

whereas equation (55) must hold along the right-running Mach line. The values for ϕ_1 and ϕ_2 on the surface are then found from

$$\phi_1(S) = \frac{C + \frac{2}{M} (\frac{\partial Z}{\partial x} + ikZ)B}{A + B} \tag{66}$$

and

$$\phi_2(S) = \frac{C - \frac{2}{M} (\frac{\partial Z}{\partial x} + ikZ)B}{A + B} \tag{67}$$

The pressure distribution over the vibrating cylindrical shell is obtained from the linearized pressure amplitude, equation (38a), and the generalized aerodynamic force Q_{mr} is found from

$$Q_{mr} = \frac{1}{2} \int_0^1 \overline{C}_{p_m}(x) \Psi_r(x)dx \tag{68}$$

where

$$Z(x) = \sum_m A_m \Psi_m(x) \tag{69}$$

and $\overline{C}_{pm}(x)$ is the aerodynamic pressure amplitude due to the deflection $\Psi_m(x)$.

These equations were programmed in Fortran IV. The program description appears in Section IV-A.

B. CASCADE

In Section II equations (28) - (30) were developed as the governing system of partial differential equations for the analysis of the cascade problem. A solution of these equations can be obtained using the method of characteristics. The following development of this method is largely based on the previous work by Teipel (Ref. 3) and Chalkley (Ref. 9), where further details can be found.

Transforming the system into a new coordinate system, $\xi(x,y)$ and $\eta(x,y)$, and considering now a system of three simultaneous equations in unknowns U_ξ, V_ξ, and C_ξ, the equations may be written

$$\xi_x U_\xi + \cot\alpha\ \xi_y V_\xi + M^2 \xi_x C_\xi = - U_\eta \eta_y - \cot\alpha\ V_\eta \eta_y - M^2 C_\eta \eta_x - ikM^2 C \tag{70}$$

$$\xi_x U_\xi + \xi_x C_\xi = - U_\eta \eta_x - C_\eta \eta_x - ikU \tag{71}$$

$$\xi_x V_\xi + \tan\alpha\ \xi_y C_\xi = - V_\eta \eta_x - \tan\alpha\ C_\eta \eta_y - ikV \tag{72}$$

If U_ξ, V_ξ, and C_ξ are made indeterminate across lines of η = constant, the determinant of the matrix of coefficients of the above equations must equal zero, or

$$- \xi_x^3 - \xi_x [\xi_y^2 - M^2\ \xi_x^2] = 0 \tag{73}$$

The solutions to this equation give the three characteristic directions in the physical (x,y) plane:

$$\frac{dy}{dx} = 0 \tag{74}$$

$$\frac{dy}{dx} = \pm \tan\alpha \tag{75}$$

In order to find the compatibility relations of the first derivatives of U, V, and C along the characteristics, equations (70) - (72) are written along lines of ξ = constant, η = constant, and along streamlines, that is, for

$$\text{str}(x,y) = \text{constant}, \quad \frac{\partial y}{\partial x} = 0 \tag{76}$$

$$\xi(x,y) = \text{constant}, \quad \frac{\partial y}{\partial x} = \frac{1}{\sqrt{M^2-1}} \tag{77}$$

$$\eta(x,y) = \text{constant}, \quad \frac{\partial y}{\partial x} = \frac{-1}{\sqrt{M^2-1}} \tag{78}$$

If an arbitrary function of x and y in the equations is denoted by f then, $\frac{\partial f}{\partial x}$ holding str constant becomes

$$\left(\frac{\partial f}{\partial x}\right)_{\text{str}} = \frac{\partial f}{\partial x} \tag{79}$$

Likewise, $\frac{\partial f}{\partial x}$ holding ξ constant becomes,

$$\left(\frac{\partial f}{\partial x}\right)_{\xi} = \frac{\partial f}{\partial x} + \frac{1}{\sqrt{M^2-1}} \frac{\partial f}{\partial y} \tag{80}$$

and, $\frac{\partial f}{\partial x}$ holding η constant becomes

$$\left(\frac{\partial f}{\partial x}\right)_{\eta} = \frac{\partial f}{\partial x} = \frac{1}{\sqrt{M^2-1}} \frac{\partial f}{\partial y} \tag{81}$$

Combining these equations and rewriting,

$$\frac{\partial f}{\partial x} = \left(\frac{\partial f}{\partial x}\right)_{\text{str}} \tag{82}$$

$$\frac{\partial f}{\partial x} = \frac{1}{2}\left[\left(\frac{\partial f}{\partial x}\right)_{\xi} + \left(\frac{\partial f}{\partial x}\right)_{\eta}\right] \tag{83}$$

$$\frac{\partial f}{\partial y} = \frac{1}{2}\sqrt{M^2-1}\left[\left(\frac{\partial f}{\partial x}\right)_{\xi} - \left(\frac{\partial f}{\partial x}\right)_{\eta}\right] \tag{84}$$

Substituting these relations into the equations of motion gives

$$\frac{1}{2}\left[\left(\frac{\partial U}{\partial x}\right)_{\xi} + \left(\frac{\partial U}{\partial x}\right)_{\eta}\right] + \frac{1}{2}\left(M^2-1\right)\left[\left(\frac{\partial V}{\partial x}\right)_{\xi} - \left(\frac{\partial V}{\partial x}\right)_{\eta}\right]$$

$$+ \frac{1}{2} M^2\left[\left(\frac{\partial C}{\partial x}\right)_{\xi} + \left(\frac{\partial C}{\partial x}\right)_{\eta}\right] + ikM^2C = 0 \qquad (85)$$

$$\frac{1}{2}\left[\left(\frac{\partial U}{\partial x}\right)_{\xi} + \left(\frac{\partial U}{\partial x}\right)_{\eta}\right] + \frac{1}{2}\left[\left(\frac{\partial C}{\partial x}\right)_{\xi} + \left(\frac{\partial C}{\partial x}\right)_{\eta}\right] + ikU = 0 \qquad (86)$$

$$\left[\left(\frac{\partial V}{\partial x}\right)_{\xi} + \left(\frac{\partial V}{\partial x}\right)_{\eta}\right] + \frac{1}{2}\left[\left(\frac{\partial C}{\partial x}\right)_{\xi} - \left(\frac{\partial C}{\partial x}\right)_{\eta}\right] + ikV = 0 \qquad (87)$$

Subtracting equation (86) from equation (85) gives an equation that is first added to equation (87) and then subtracted from equation (87). The result is two compatibility relations that must be satisfied along lines of constant ξ and lines of constant η, i.e.,

$$\left(\frac{\partial V}{\partial x}\right)_{\xi} + \left(\frac{\partial C}{\partial x}\right)_{\xi} + ik\left[V + \frac{1}{M^2-1}\left(M^2C-U\right)\right] = 0 \qquad (88)$$

$$\left(\frac{\partial V}{\partial x}\right)_{\eta} - \left(\frac{\partial C}{\partial x}\right)_{\eta} + ik\left[V - \frac{1}{M^2-1}\left(M^2C-U\right)\right] = 0 \qquad (89)$$

The third compatibility relation is obtained by considering equation (86) along a streamline

$$\left(\frac{\partial U}{\partial x}\right)_{str} + \left(\frac{\partial C}{\partial x}\right)_{str} + ikU = 0 \qquad (90)$$

If irrotational flow is assumed, the equation of irrotationality

$$\frac{\partial u}{\partial y} - \frac{\partial v}{\partial x} = 0 \qquad (91)$$

may be substituted for the second equation of motion in the system. With the assumption of simple harmonic motion, it may be written in terms of Teipel's amplitude functions,

$$\frac{\partial U}{\partial y} - \sqrt{M^2-1}\,\frac{\partial V}{\partial x} = 0 \qquad (92)$$

The non-dimensional system of equations can then be written as,

$$\frac{\partial U}{\partial x} + \sqrt{M^2-1}\,\frac{\partial V}{\partial y} + M^2\,\frac{\partial C}{\partial x} + ikM^2C = 0 \qquad (93)$$

$$\frac{\partial U}{\partial x} + \frac{\partial C}{\partial x} + ikU = 0 \qquad (94)$$

$$\frac{\partial U}{\partial y} - \sqrt{M^2-1}\,\frac{\partial V}{\partial x} = 0 \qquad (95)$$

By transforming equations (93) - (95) to the arbitrary coordinate system, ξ and η, the system can be considered one of simultaneous equations in U_ξ, V_ξ, and C_ξ. By making any of these derivatives indeterminate, the results of setting the determinant of the matrix of their coefficients equal to zero are equations (74) and (75), the characteristic directions.

By writing the equations of motion, as before, along lines of ξ = constant and η = constant and performing the same algebraic manipulations, the two compatibility relations can be determined:

$$\left(\frac{\partial U}{\partial x}\right)_\xi - \left(\frac{\partial V}{\partial x}\right)_\xi + ik\,\frac{M^2}{M^2-1}\,(U-C) = 0 \qquad (96)$$

$$\left(\frac{\partial U}{\partial x}\right)_\eta + \left(\frac{\partial V}{\partial x}\right)_\eta + ik\,\frac{M^2}{M^2-1}\,(U-C) = 0 \qquad (97)$$

The third compatibility relation is, as before,

$$\left(\frac{\partial U}{\partial x}\right)_{str} + \left(\frac{\partial C}{\partial x}\right)_{str} + ikU = 0 \qquad (90)$$

The equations of motion are now reduced to a system containing only derivatives with respect to x. It is, thus, possible to solve these equations using finite differences in the following manner: separating the real from the imaginary parts, the equations can be written,

$$\left(\frac{\partial U_R}{\partial x}\right)_{str} + \left(\frac{\partial C_R}{\partial x}\right)_{str} - kU_I = 0 \qquad (98)$$

$$\left(\frac{\partial U_I}{\partial x}\right)_{str} + \left(\frac{\partial C_I}{\partial x}\right)_{str} + kU_R = 0 \qquad (99)$$

$$\left(\frac{\partial U_R}{\partial x}\right)_{\xi} - \left(\frac{\partial V_R}{\partial x}\right)_{\xi} - k\,\frac{M^2}{M^2-1}\,(U_I - C_I) = 0 \qquad (100)$$

$$\left(\frac{\partial U_I}{\partial x}\right)_{\xi} - \left(\frac{\partial V_I}{\partial x}\right)_{\xi} + k\,\frac{M^2}{M^2-1}\,(U_R - C_R) = 0 \qquad (101)$$

$$\left(\frac{\partial U_R}{\partial x}\right)_{\eta} + \left(\frac{\partial V_R}{\partial x}\right)_{\eta} - k\,\frac{M^2}{M^2-1}\,(U_I - C_I) = 0 \qquad (102)$$

$$\left(\frac{\partial U_I}{\partial x}\right)_{\eta} + \left(\frac{\partial V_I}{\partial x}\right)_{\eta} + k\,\frac{M^2}{M^2-1}\,(U_R - C_R) = 0 \qquad (103)$$

To write these equations in finite difference form the computational molecule shown in Figure 4 will be used. With the velocity values all known at P_{11}, P_{12}, and P_{21}, all the values at P_{22} can be calculated. The distance Δx is arbitrary.

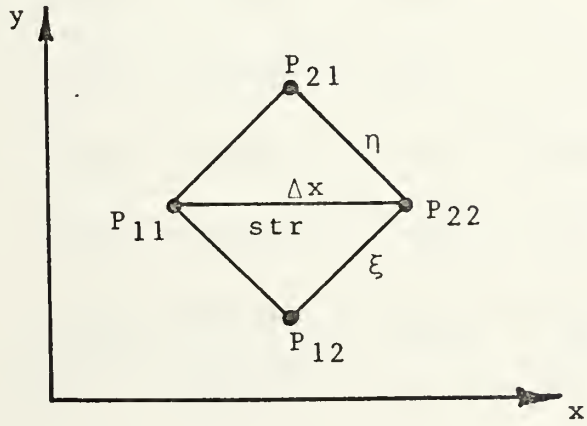


Figure 4. General Flow Field Molecule

If F is an arbitrary flow quantity, its partial derivatives can be written as

$$\left(\frac{\partial F}{\partial x}\right)_{str} = \frac{F_{22} - F_{11}}{\Delta x} \tag{104}$$

$$\left(\frac{\partial F}{\partial x}\right)_{\xi} = \frac{F_{22} - F_{12}}{\frac{1}{2}\Delta x} \tag{105}$$

$$\left(\frac{\partial F}{\partial x}\right)_{\eta} = \frac{F_{22} - F_{21}}{\frac{1}{2}\Delta x} \tag{106}$$

The values of the flow quantities in the equations are averages, such that

$$(F)_{str} = \frac{1}{2}(F_{11} + F_{22}) \tag{107}$$

$$(F)_{\xi} = \frac{1}{2}(F_{12} + F_{22}) \tag{108}$$

$$(F)_{\eta} = \frac{1}{2}(F_{21} + F_{22}) \tag{109}$$

With these relations substituted into equations (98) - (103), the system becomes after some algebraic manipulation,

$$U_{22R} + C_{22R} - A_I U_{22I} = K_{12R} \tag{110}$$

$$U_{22I} + C_{22I} + A_I U_{22R} = K_{12I} \tag{111}$$

$$U_{22R} - V_{22R} + B_I(C_{22I} - U_{22I}) = K_{34R} \tag{112}$$

$$U_{22I} - V_{22I} + B_I(U_{22R} - C_{22R}) = K_{34I} \tag{113}$$

$$U_{22R} + V_{22R} + B_I(C_{22I} - U_{22I}) = K_{56R} \tag{114}$$

$$U_{22I} + V_{22I} + B_I(U_{22R} - C_{22R}) = K_{56I} \tag{115}$$

where

$$K_{12R} = U_{11R} + C_{11R} + A_I U_{11I} \tag{116}$$

$$K_{12I} = U_{11I} + C_{11I} - A_I U_{11R} \tag{117}$$

$$K_{34R} = U_{12R} - V_{12R} + B_I(U_{12I} - C_{12I}) \tag{118}$$

$$K_{34I} = U_{12I} - V_{12I} - B_I(U_{12R} - C_{12R}) \tag{119}$$

$$K_{56R} = U_{21R} + V_{21R} + B_I(U_{21I} - C_{21I}) \tag{120}$$

$$K_{56I} = U_{21I} + V_{21I} - B_I(U_{21R} - C_{21R}) \tag{121}$$

and

$$A_I = \frac{1}{2} k\Delta x \tag{122}$$

$$B_I = \frac{1}{4} k \frac{M^2}{M^2-1} \Delta x \tag{123}$$

As shown in Teipel (Ref. 3), solving these equations gives,

$$U_{22R} = \frac{(1-A_IB_I)[\frac{1}{2}(K_{34R}+K_{56R})-B_IK_{12I}]+2B_I[\frac{1}{2}(K_{34I}+K_{56I})+B_IK_{12R}]}{(1-A_IB_I)^2 + (2B_I)^2} \tag{124}$$

$$U_{22I} = \frac{(1-A_IB_I)[\frac{1}{2}(K_{34I}+K_{56I})+B_IK_{12R}]-2B_I[\frac{1}{2}(K_{34R}+K_{56R})-B_IK_{12I}]}{(1-A_IB_I)^2 + (2B_I)^2} \tag{125}$$

$$V_{22R} = \frac{1}{2} (K_{56R} - K_{34R}) \tag{126}$$

$$V_{22I} = \frac{1}{2} (K_{56I} - K_{34I}) \tag{127}$$

$$C_{22R} = K_{12R} - U_{22R} + A_IU_{22I} \tag{128}$$

$$C_{22I} = K_{12I} - U_{22I} - A_IU_{22R} \tag{129}$$

These are the finite difference equations for a point in the general flow field.

At a point on the upper surface of a cascade blade, the normal velocity V_{22} is prescribed by the movement of the blade.

This velocity was given in the problem formulation. Here the computational molecule is shown in Figure 5.

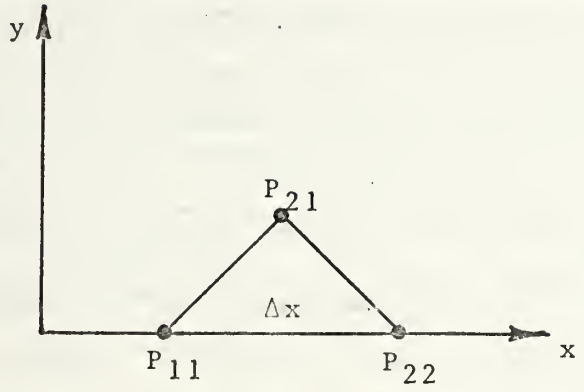


Figure 5. Upper Surface Molecule

The applicable equations are,

$$U_{22R} + C_{22R} - A_I U_{22I} = K_{12R} \tag{130}$$

$$U_{22I} + C_{22I} + A_I U_{22R} = K_{12I} \tag{131}$$

$$U_{22R} - B_I U_{22I} + B_I C_{22I} = K_{56R} - K_{34R} \tag{132}$$

$$U_{22I} + B_I U_{22R} - B_I C_{22R} = K_{56I} - K_{34I} \tag{133}$$

where K_{12} and K_{56} are as before, but,

$$K_{34R} = V_{22R} \tag{134}$$

$$K_{34I} = V_{22I} \tag{135}$$

as given by the boundary conditions. Solving for the flow quantities gives

$$U_{22R} = \frac{(1-A_I B_I)[K_{56R}-K_{34R}-B_I K_{12I}]+2B_I[K_{56I}-K_{34I}+B_I K_{12R}]}{(1-A_I B_I)^2 + (2B_I)^2} \tag{136}$$

$$U_{22I} = \frac{(1-A_I B_I)[K_{56I}-K_{34I}+B_I K_{12R}]-2B_I[K_{56R}-K_{34R}-B_I K_{12I}]}{(1-A_I B_I)^2 + (2B_I)^2} \tag{137}$$

$$C_{22R} = K_{12R} - U_{22R} + A_I U_{22I} \tag{138}$$

$$C_{22I} = K_{12I} - U_{22I} - A_I U_{22R} \tag{139}$$

and V_{22R} and V_{22I} are known from the boundary conditions. These are the finite difference equations for the flow velocities on the upper surface of a cascade blade.

Likewise, at the lower surface of a cascade blade, the normal velocity, V_{22} is prescribed by the movement of the blade, and was given in the problem formulation. The computational molecule, different from that at the upper surface of the blade, is shown in Figure 6.

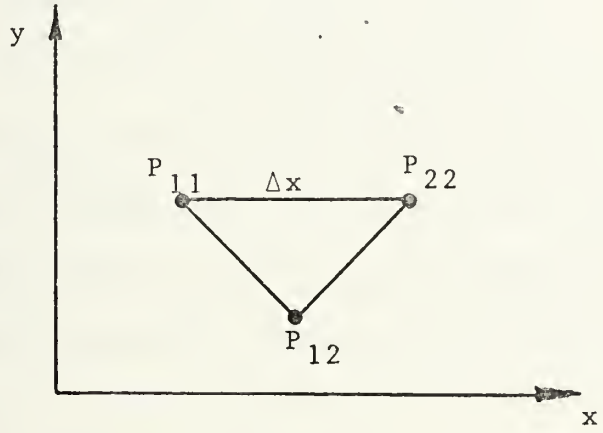


Figure 6. Lower Surface Molecule

Here, the applicable equations are,

$$U_{22R} + C_{22R} - A_I U_{22I} = K_{12R} \tag{140}$$

$$U_{22I} + C_{22I} + A_I U_{22R} = K_{12I} \tag{141}$$

$$U_{22R} - B_I U_{22I} + B_I C_{22I} = K_{56R} + K_{34R} \tag{142}$$

$$U_{22I} + B_I U_{22R} - B_I C_{22R} = K_{56I} + K_{34I} \tag{143}$$

where K_{12} and K_{34} are as originally stated, but,

$$K_{56R} = V_{22R} \tag{144}$$

$$K_{56I} = V_{22I} \tag{145}$$

as given by the boundary conditions. Thus, the finite difference equations for the flow velocities at the lower surface of a cascade blade are,

$$U_{22R} = \frac{(1-A_I B_I)[K_{56R}+K_{34R}-B_I K_{12I}]+2B_I[K_{56I}+K_{34I}+B_I K_{12R}]}{(1-A_I B_I)^2 + (2B_I)^2} \tag{146}$$

$$U_{22I} = \frac{(1-A_I B_I)[K_{56I}+K_{34I}+B_I K_{12R}]-2B_I[K_{56R}+K_{34R}-B_I K_{12I}]}{(1-A_I B_I)^2 + (2B_I)^2} \tag{147}$$

$$C_{22R} = K_{12R} - U_{22R} + A_I U_{22I} \tag{148}$$

$$C_{22I} = K_{12I} - U_{22I} - A_I U_{22R} \tag{149}$$

where V_{22R} and V_{22I} are given from the boundary condition.

To obtain the conditions on the initial left-running Mach line, assume that P_{11} and P_{21} as shown in Figure 4 are just in the free-stream and then let Δx shrink to zero. Since $K_{12} = K_{56} = 0$ and A_I and B_I go to zero as Δx goes to zero, the initial finite difference equations for ξ=constant are,

$$U_{22} + C_{22} = 0 \tag{150}$$

$$U_{22} - V_{22} + V_{12} = 0 \tag{151}$$

$$U_{22} + V_{22} = 0 \tag{152}$$

thus,

$$U_{22} = - V_{22} = - C_{22} \tag{153}$$

Equation (96) can be written,

$$\left(\frac{\partial U}{\partial x}\right)_\xi = - ikU \tag{154}$$

Integrating gives,

$$U = U|_{x=0} e^{-ik\frac{M^2}{M^2-1}x} \tag{155}$$

thus,

$$U_{22R} = -V_{22R}(0)\cos\left(k\frac{M^2}{M^2-1}x\right) - V_{22I}(0)\sin\left(k\frac{M^2}{M^2-1}x\right) \tag{156}$$

$$U_{22I} = -V_{22I}(0)\cos\left(k\frac{M^2}{M^2-1}x\right) + V_{22R}(0)\sin\left(k\frac{M^2}{M^2-1}x\right) \tag{157}$$

$$V_{22R} = -U_{22R} \tag{158}$$

$$V_{22I} = -U_{22I} \tag{159}$$

$$C_{22R} = -U_{22R} \tag{160}$$

$$C_{22I} = -U_{22I} \tag{161}$$

For the initial right-running Mach line, P_{11} and P_{12} of Figure 4 are assumed to be just in the free-stream and Δx is brought to zero. Since $K_{12} = K_{34} = 0$, and $A_I = B_I = 0$,

$$U_{22} + C_{22} = 0 \tag{162}$$

$$U_{22} - V_{22} = 0 \tag{163}$$

$$U_{22} - U_{21} + V_{22} = 0 \tag{164}$$

thus,

$$U_{22} = V_{22} = -C_{22} \tag{165}$$

Equation (97) can then be written,

$$\left(\frac{\partial U}{\partial x}\right)_\eta = -ikU \tag{166}$$

Integrating gives,

$$U = V(0)e^{-ik\frac{M^2}{M^2-1}x} \tag{167}$$

Thus,

$$U_{22R} = V_{22R}(0)\cos\left(k\frac{M^2}{M^2-1}x\right)+V_{22I}(0)\sin\left(k\frac{M^2}{M^2-1}x\right) \quad (168)$$

$$U_{22I} = V_{22I}(0)\cos\left(k\frac{M^2}{M^2-1}x\right)-V_{22R}(0)\sin\left(k\frac{M^2}{M^2-1}x\right) \quad (169)$$

$$V_{22R} = U_{22R} \quad (170)$$

$$C_{22R} = -U_{22R} \quad (171)$$

$$V_{22I} = U_{22I} \quad (172)$$

$$C_{22I} = -U_{22I} \quad (173)$$

At the first position encountered on a blade other than the first blade there are two grid points for which the flow field quantities must be computed; one on the upper surface and one on the lower surface of the blade.

As in Figure 7 the blade is assumed to protrude a very small distance past P_{22}, thus necessitating the computation of the upper and lower flow velocities at this point. In this case, the equations developed earlier for an upper surface grid point are used for P_{22U} and those developed for a lower surface grid point are used at P_{22L}.

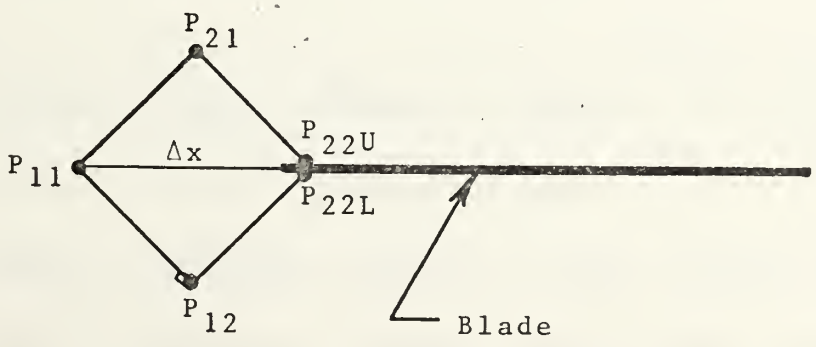


Figure 7. New Blade Molecule

At a position just aft of a blade there are also two grid points to be accounted for; one above and one below the slip plane in the wake. As in Figure 8 the point P_{22} is assumed to be a very small distance aft of the blade's trailing edge.

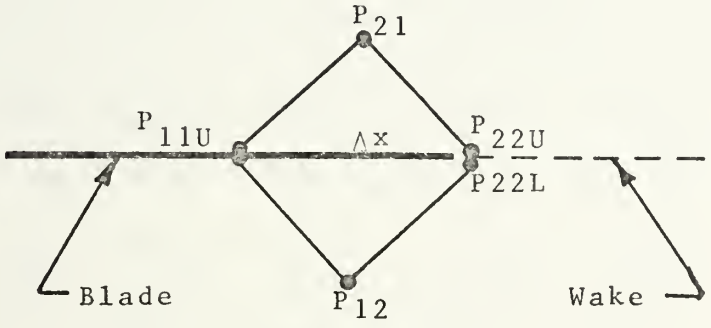


Figure 8. Wake Molecule

At this position the equations for an upper blade grid point apply to P_{22U} and those for a lower grid point apply to P_{22L}, i.e.,

$$U_{22RU} + C_{22RU} - A_I U_{22IU} = K_{12RU} \quad (174)$$

$$U_{22IU} + C_{22IU} + A_I U_{22RU} = K_{12IU} \quad (175)$$

$$U_{22RU} + B_I(C_{22IU} - U_{22IU}) = K_{56R} - V_{22RU} \quad (176)$$

$$U_{22IU} + B_I(U_{22RU} - C_{22RU}) - K_{56I} - V_{22IU} \quad (177)$$

and

$$U_{22RL} + C_{22RL} - A_I U_{22IL} = K_{12RL} \quad (178)$$

$$U_{22IL} + C_{22IL} + A_I U_{22RL} = K_{12IL} \quad (179)$$

$$U_{22RL} + B_I(C_{22IL} - U_{22IL}) = V_{22RL} + K_{34R} \quad (180)$$

$$U_{22IL} + B_I(U_{22RL} - C_{22RL}) = V_{22IL} + K_{34I} \quad (181)$$

where

$$K_{12RU} = U_{11RU} + C_{11RU} + A_I U_{11IU} \tag{182}$$

$$K_{12IU} = U_{11IU} + C_{11IU} - A_I U_{11RU} \tag{183}$$

$$K_{12RL} = U_{11RL} + C_{11RL} + A_I U_{11IL} \tag{184}$$

$$K_{12IL} = U_{11IL} + C_{11IL} - A_I U_{11RL} \tag{185}$$

$$K_{34R} = U_{12R} - V_{12R} + B_I(U_{12I} - C_{12I}) \tag{186}$$

$$K_{34I} = U_{12I} - V_{12I} - B_I(U_{12R} - C_{12R}) \tag{187}$$

$$K_{56R} = U_{21R} + V_{21R} + B_I(U_{21I} - C_{21I}) \tag{188}$$

$$K_{56I} = U_{21I} + V_{21I} - B_I(U_{21R} - C_{21R}) \tag{189}$$

Equations (174) - (181) form a system of eight equations with twelve unknowns. However, since it is known that the wake cannot support a pressure difference and that the normal velocities on either side of the wake must be equivalent, i.e.,

$$V_{22RU} = V_{22RL} \equiv V_{22R} \tag{190}$$

$$V_{22IU} = V_{22IL} \equiv V_{22I} \tag{191}$$

$$C_{22RU} = C_{22RL} \equiv C_{22R} \tag{192}$$

$$C_{22IU} = C_{22IL} \equiv C_{22I} \tag{193}$$

we can eliminate four of the unknowns and achieve a determinate system of equations. Equations (174) - (181) become

$$U_{22RU} + C_{22R} - A_I U_{22IU} = K_{12RU} \tag{194}$$

$$U_{22IU} + C_{22I} + A_I U_{22RU} = K_{12IU} \tag{195}$$

$$U_{22RU} + V_{22R} + B_I(C_{22I} - U_{22IU}) = K_{56R} \tag{196}$$

$$U_{22IU} + V_{22I} + B_I(U_{22RU} - C_{22R}) = K_{56I} \tag{197}$$

$$U_{22RL} + C_{22R} - A_I U_{22IL} = K_{12RL} \tag{198}$$

$$U_{22IL} + C_{22I} + A_I U_{22RL} = K_{12IL} \tag{199}$$

$$U_{22RL} - V_{22R} + B_I(C_{22I} - U_{22IL}) = K_{34R} \tag{200}$$

$$U_{22IL} - V_{22I} + B_I(U_{22RL} - C_{22R}) = K_{34I} \tag{201}$$

The pressure distributions along the surfaces of the blades can be determined from equation (11). Dividing both sides of this equation by p_∞ and using Teipel's amplitude function for c, the equation can be written,

$$P(x,y) = \gamma M^2 C(x,y)\big|_{y=0^+,\ y=0^-} \tag{202}$$

where

$$P(x,y)e^{i\omega t} = \frac{p-p_\infty}{p_\infty} \tag{203}$$

Thus the non-dimensional pressure distributions along the upper and lower surfaces of the blades in a cascade can be computed in a point by point calculation of U, V, and C along lines of ξ=constant and η=constant.

These equations were programmed in Fortran IV. The program description appears in Section IV-B.

C. THE LINEARIZED METHOD OF CHARACTERISTICS OF SAUER AND HEINZ

Sauer and Heinz (Ref. 11) developed a linearized method of characteristics for supersonic flow past bodies of revolution at zero angle of attack. This method also is derived from the linearized potential equation for steady supersonic flow, i.e.,

$$(M^2-1)\phi_{xx} - \phi_{rr} - \frac{1}{r}\phi_r = 0 \tag{204}$$

Sauer and Heinz showed that the compatibility relations can be written as

$$\frac{\partial (rv)}{\partial \xi} = -\sqrt{M^2-1}\, r\, \frac{\partial u}{\partial \xi} \tag{205}$$

and

$$\frac{\partial (rv)}{\partial \eta} = \sqrt{M^2-1}\, r\, \frac{\partial u}{\partial \eta} \tag{206}$$

where

$$u = \frac{\partial \phi}{\partial x} \tag{207}$$

$$v = \frac{\partial \phi}{\partial r} \tag{208}$$

and

$$\xi = x - \sqrt{M^2-1}\, r \tag{209}$$

$$\eta = x + \sqrt{M^2-1}\, r \tag{210}$$

are the characteristic coordinates as explained previously.

Writing this equation in finite difference form one obtains for u and v at point 4 (Figure 9) using the known values in points 2 and 3,

$$U_4 = \frac{(rv)_3 - (rv)_2 + \sqrt{M^2-1}\,(r_\eta u_3 + r_\xi u_2}{\sqrt{M^2-1}\,(r_\xi + r_\eta)} \tag{211}$$

$$(vr)_4 = \frac{r_\xi (rv)_3 + r_\eta (rv)_2 - \sqrt{M^2-1}\, r_\xi r_\eta (u_2 - u_3)}{r_\xi + r_\eta} \tag{212}$$

where r_η and r_ξ are the average r coordinates between 3 and 4 and between 2 and 4 respectively. Computing the flow properties at the body using the exact flow tangency condition leads to

$$u_5 = \frac{\sqrt{M^2-1}\, r_\eta u_4 + (rv)_4 - r_5 \left(\frac{dh}{dx}\right)_5}{\sqrt{M^2-1}\, r_\eta + r_5 \left(\frac{dh}{dx}\right)_5} \tag{213}$$

$$v_5 = \left(\frac{dh}{dx}\right)_5 (1+u_5) \tag{214}$$

where $r = h(x)$ describes the body contour.

However, in order to apply the Sauer-Heinz procedure to the case of an infinitely long cylinder with a wavy surface between $0<x<1$ a different starting procedure is needed. It is easily seen from Figure 9 that

$$r_2 = \frac{h_0}{2} \left(\frac{1}{\sqrt{M^2-1}\, \tan \delta_0} + 1\right) \tag{215}$$

realizing that

$$r_\eta = \frac{R_1 + R_2}{2} \tag{216}$$

where

$$R_1 = R_0 + r_1 \tag{217}$$

and

$$R_2 = R_0 + h_0 \tag{218}$$

one obtains

$$r_\eta = R + \frac{h_0}{4} \left(3 + \frac{1}{\sqrt{M^2-1}\, \tan \delta_0}\right) \tag{219}$$

Since there is no disturbance upstream of the leading edge Mach line, one obtains from equation (213)

$$U_2 = \frac{-(R+h_0) \tan \delta_0}{\sqrt{M^2-1}\, r_\eta + (R+h_0) \tan \delta_0} \tag{220}$$

where the point 2 is the first body point.

Finally, the pressure coefficient is obtained from

$$C_p - 2u - v^2 + u^2 \quad (M^2-1) \tag{221}$$

The program listing for these equations is given in Appendix I.

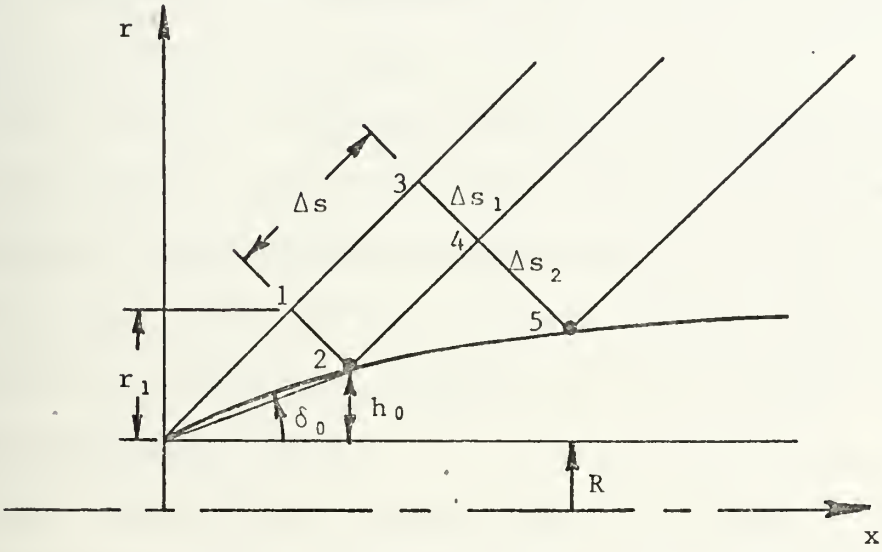


Figure 9. Sauer-Heinz Characteristic Network

IV. COMPUTATIONAL PROCEDURE

A. CYLINDRICAL SHELL COMPUTER PROGRAM

This program calculates the pressure distribution over a harmonically vibrating cylindrical shell. It is valid only for a cylinder in a supersonic flow field vibrating with small amplitude vibrations.

The program calculates the flow field properties adjacent to the cylinder using the method of characteristics finite difference equations developed previously in Section III. The pressure distribution is then integrated using the trapezoidal rule according to equation (68) to obtain the generalized aerodynamic force.

The computational diagram shown in Figure 3 is used to calculate the flow field quantities at S in the general flow field and on the cylinder surface. The distance Δs shown in the figure is determined by

$$\Delta s = \frac{L \cos \alpha}{\nu} \tag{222}$$

where ν is the grid fineness ratio. The grid fineness ratio is an arbitrary input variable into the program and is equal to one less than the total number of grid points on the cylinder surface.

Input parameters are entered into the program on two data cards. The first card contains the date, e.g., 11 APR 1973, in the first twelve columns. The second card contains the following input data:

SPACE	DATA
1-20	Free-stream Mach number, M
21-30	Radius to length ratio, R
31-40	Reduced frequency, k
41-45	Axial mode number, m
46-50	r, as defined in equation (68) for the generalized aerodynamic force.
51-55	Circumferential mode number, n
56-60	Grid fineness ratio, ν

where the first three quantities are floating point numbers and the rest are integers. The definitions of these parameters are given again in the alphabetical listing of all program variables in Appendix A.

The program has five subroutines. They are as follows:

COMPXR	Computes the x and r position of the given grid point.
MACHLN	Computes the flow quantities along the initial left-running Mach line at the given grid point.
GENFPT	Computes the flow quantities at a general flow field grid point.
PANPNT	Computes the flow quantities on the cylinder surface at the given grid point.
INTEG	Computes the generalized aerodynamic force.

Flow quantities along the initial left-running Mach line are computed using the equations derived in the method of characteristics for initial conditions. Those quantities on the cylinder surface are computed using the equations derived

for the boundary condition. The values of the velocity potential and its derivatives are derived as a result of the analysis of the flow tangency condition and have been given previously. The procedure starts at the initial Mach line and continues down successive right-running Mach lines. The pressure coefficient is computed when the cylinder surface is met and stored in an array for later use in the integration of the generalized aerodynamic force.

A flow diagram of this program is given in Appendix B.

B. OSCILLATING CASCADE PROGRAM

This program calculates the pressure distribution over the upper and lower surfaces of the blades of a subsonic leading edge locus cascade which is oscillating in pitch at arbitrary frequency and interblade phase angle. It is valid only for a cascade in a supersonic flow field vibrating with small amplitude vibrations. The program calculates the flow field quantities within the area governed by the aft-running Mach lines emanating from the leading edge of the first blade and the forward-running Mach lines emanating from the trailing edge of the last blade of the cascade (see Figure 10). These flow field properties are calculated using the method of characteristics finite difference equations developed previously in Section III-B.

The computational molecules shown in Figures 4, 5, 6, 7, and 8 are used to calculate the flow field properties in the general flow field, upper surfaces, lower surfaces, leading edge of a staggered blade, and the wake aft of a blade,

respectively. The distance Δx shown in the figures is determined by

$$\Delta x = \frac{2d}{\nu} \cot \alpha \qquad (223)$$

where ν is the grid fineness ratio. This parameter is an arbitrary input variable into the program and is equal to one less than the total number of grid points on any Mach line that runs from one blade to the next.

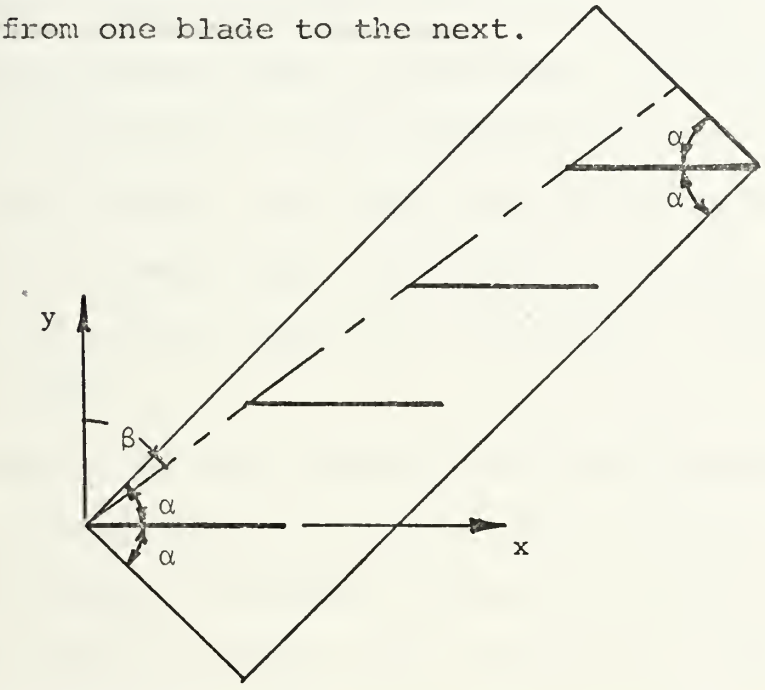


Figure 10. Cascade Computational Area

Input parameters are entered into the program by means of the NAMELIST option of Fortran. In using this option, input parameters are combined in a titled list and then referred to in the program by this title (NAM1). The real advantage of using NAMELIST is that each data card need only contain the value of one input variable while the format of the data card is such that the variable's name is punched on the card as well as its value. In this way, when a different value for an input variable is desired, rather than changing the

entire data deck, only one easily identifiable data card need be changed.

The format for a NAMELIST data deck is as follows: The first column in each card of the data deck is left blank. The first card starts in the second column with the symbol & (ampersand) followed immediately by the title of the NAMELIST (here, NAM1). The next cards list the variables and their input values, one per card, in any order. The format here, starting in the second column and followed by a comma, is: variable name = value. The last card in the data deck and the signal to the computer that the NAMELIST has ended again starts in the second column with the symbol & immediately followed by END.

The value of an input variable may be written in any format as long as integer variables are written without decimal points and real variables are written with a decimal point. Blanks are taken as zeroes, but a comma must appear somewhere between a desired input value and the next data card. The date of the run is entered on a data card in the first twelve spaces. This data card is inserted prior to the NAMELIST dataset.

To illustrate the input procedure, the following example is proposed:

$\nu = 100$ $\quad$ $d = 0.8$

$M = \sqrt{2}$ $\quad$ $\beta = 55°$

$k = 0.75$ $\quad$ $\delta = 45°$

$x_0 = 0.5$

5 blades

The data cards needed are as follows:

CARD	FORMAT	DESCRIPTION
1	22 JUNE 1973	Date
2	&NAM1	NAMELIST title
3	NGRDFN=100,	Grid fineness ratio, must be an even integer, subject to the constraint of equation (227).
4	FSTRMN=1.414214,	Free-stream Mach number
5	REDFRQ=0.80,	Reduced frequency
6	XSUBO=0.50,	Elastic axis position
7	TNWDST=0.80,	Vertical distance between blades.
8	STGANG=55.0,	Complimentary stagger angle, $\tan \beta > \cot \alpha$.
9	FAZE=45.0,	Interblade phase angle
10	NUMBLD=5,	Number of blades
11	&END	End of NAMELIST

The definitions of these parameters are given again in the alphabetical listing of all program variables in Appendix C.

The program has nine subroutines. They are as follows:

INPUT — Reads in all input data and sets up the finite difference grid.

INTIAL — Initializes all the flow quantities at the leading edge of the first blade and initializes most of the logic variables.

MACHLN — Computes the flow quantities along the initial left and right-running Mach lines at the given grid point.

TOP — Computes the flow quantities on the upper surface of a blade at the given grid point.

GENFPT — Computes the flow quantities at the given general flow field point.

BOTTOM — Computes the flow quantities on the lower surface of a blade at the given grid point.

NEWBLD — Computes the flow quantities at the leading edge of all staggered blades.

WAKE — Computes the flow quantities in the wake aft of each blade at the given grid point. This subroutine uses IBM subroutine SIMQ for the solution of its simultaneous equations.

COMPXY — Computes the x and y position of the given grid point.

In INPUT the grid is determined using the grid fineness ratio, blade spacing, number of blades, and the free-stream Mach number. The value of Δx as shown in equation (223) is obtained. The stagger angle is then adjusted such that the distance the upper blades are staggered back is the nearest multiple number of Δx increments. The "compatible stagger angle" is then printed out. The blades may be staggered such that the first staggered blade is at least one Δx increment aft of the initial left-running Mach line, but no further forward, or,

$$d \cdot (\tan \beta - \cot \alpha) \geq \Delta x \qquad (224)$$

Flow quantities along the initial left and right-running Mach lines are computed using the equations derived in the method of characteristics, Section B, for initial conditions. Those flow quantities along the blades' surfaces are computed using the equations derived for the appropriate boundary condition. The equations for the normal velocities are derived as a result of the analysis of the flow tangency condition and have been given previously. The flow quantities

in the wake aft of each blade are computed using the set of simultaneous equations derived as a result of the assumptions made about the flow quantities on either side of the wake and are given in the Method of Characteristics, Section III-B.

The procedure starts at the leading edge of the first blade and computes the flow quantities at grid points along the initial right-running Mach line. It then jumps back to the initial left-running Mach line and computes the flow quantities along the next right-running Mach line, using the applicable subroutines at the appropriate grid points. Only flow quantities on the Mach line presently being computed and the Mach line which preceeded it are kept in storage at any one time. Flow quantities at successive grid points are computed in like manner until the trailing edge of the last blade of the cascade is met.

A flow diagram of this program is given in Appendix D.

C. SAUER-HEINZ COMPUTER PROGRAM

This program computes the pressure distribution over a cylindrical shell or body of revolution. It is written so that the shape of the body or cylindrical surface may be easily changed. It is valid only for non-oscillating bodies or shells in a supersonic flow field at zero angle of attack. The following discussion applies to the cylindrical shell.

The program calculates the flow field adjacent the cylinder using the method of characteristics finite difference equations developed previously in Section III-C.

The computational diagram shown in Figure 9 is used to compute the flow field properties at 4 in the general flow field and at 5 on the cylinder surface. The distance Δs shown in the figure is determined by,

$$\Delta s = \frac{\cos \alpha}{\nu} \tag{225}$$

where ν is the grid fineness ratio. The grid fineness ratio is an arbitrary input variable into the program and is equal to one less than the number of grid points on the initial Mach line. The length Δs_n, n=1,2,3... is determined by the intersection of the right-running Mach lines with the surface.

Input parameters are entered into the program on one data card. This card contains the following input data:

SPACES	DATA
1-10	Free-stream Mach number, M
11-20	Maximum amplitude of the axial mode, h_0
21-30	Cylinder radius to length ratio, R.
31-40	Axial mode number, m
41-50	Grid fineness ratio, ν

where the first three quantities are floating point numbers and the last two are integers. The definitions of these quantities are given again in the alphabetical listing of all program variables in Appendix E.

The program has three subroutines. They are as follows:

COMPXR	Computes the x and r position of the grid point and determines the position of the intersection of the current right-running Mach line with the body.
GENFPT	Computes the flow quantities at a general flow field grid point.

SURFAC — Computes the flow quantities at the given surface point.

Flow quantities on the initial left-running Mach line are assumed to be zero. Those quantities on the surface are computed using the appropriate equations developed in Section III-C, for surface grid points.

The procedure starts at the initial Mach line and continues down successive right-running Mach lines. The pressure coefficient is computed when the surface is met.

A flow diagram of this program is given in Appendix F.

V. RESULTS AND COMPARISONS

A. CYLINDRICAL SHELL

Pressure distributions and generalized aerodynamic forces were computed for sinusoidal deflection modes, i.e.,

$$\Psi_m(x) = \sin(m\pi x) \qquad m=1,2,... \qquad (226)$$

Figures 11 through 16 show results for the generalized aerodynamic forces Q_{11} and Q_{12} as a function of circumferential mode number for two radius to length ratios, r=0.5 and r=1.0, at a Mach number of $\sqrt{2}$ and reduced frequencies of zero and one. Also shown is the comparison with two previous solutions by Dowell and Widnall (Ref. 2) and by Matsuzaki and Kobayashi (Ref. 5). The Dowell and Widnall solution is also based on the complete linearized unsteady potential equation, but uses Laplace transform techniques for its solution. As can be seen, agreement between this solution and the method of characteristics solution is excellent. For the sake of interest, a recent approximate solution by Matsuzaki and Kobayashi, valid only for large values of the circumferential mode number is also shown.

Figures 17 through 24 show the generalized aerodynamic forces Q_{11R} and Q_{12R} as a function of radius to length ratio again for a Mach number of $\sqrt{2}$, zero reduced frequency, and circumferential mode numbers n=0,1,2, and 3. The main purpose of these figures is to show that the infinite cylinder theory is a poor approximation for small radius to length ratios, but approaches the exact theory for large values of R.

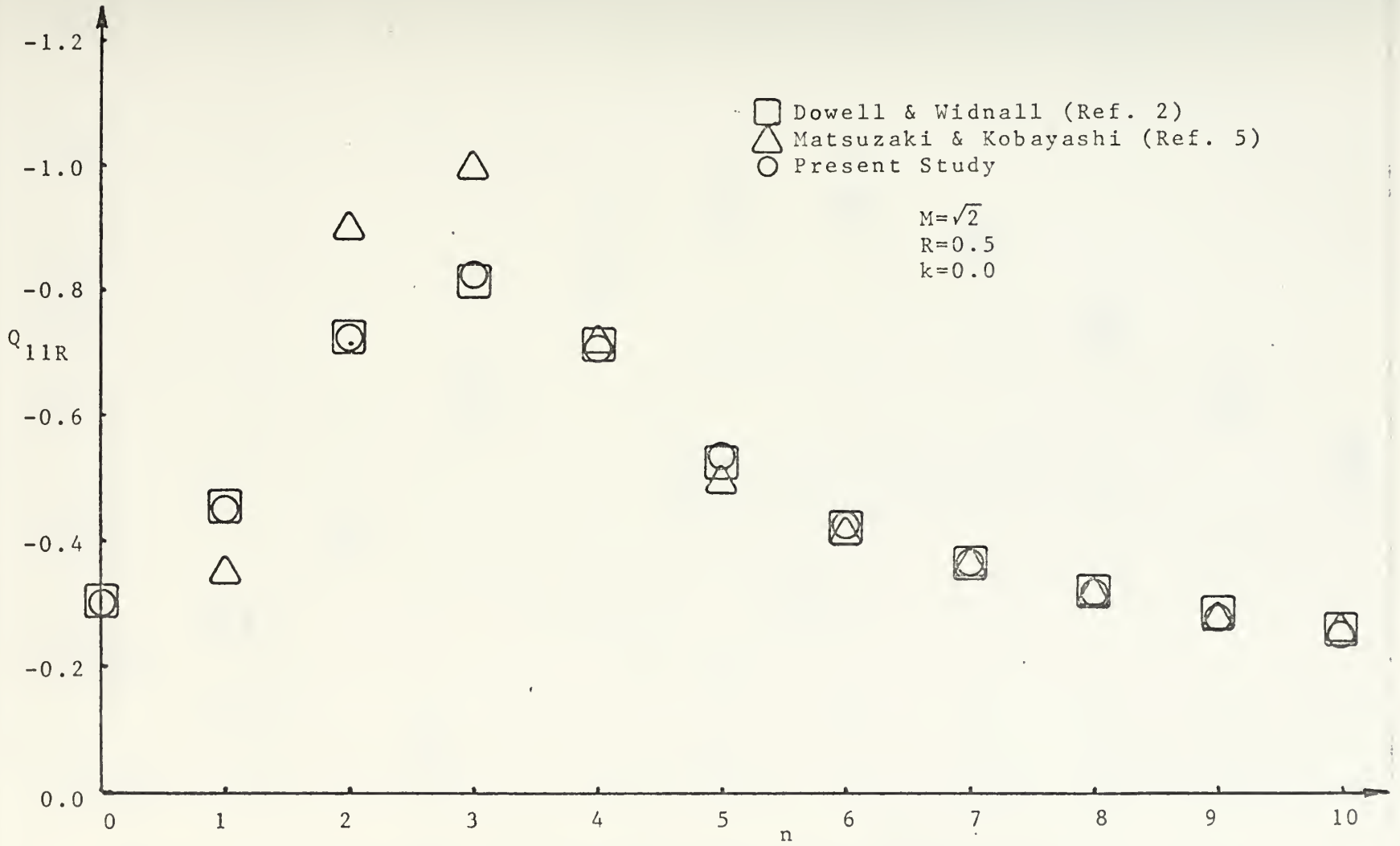


Figure 11. Generalized Aerodynamic Force vs. Circumferential Mode Number

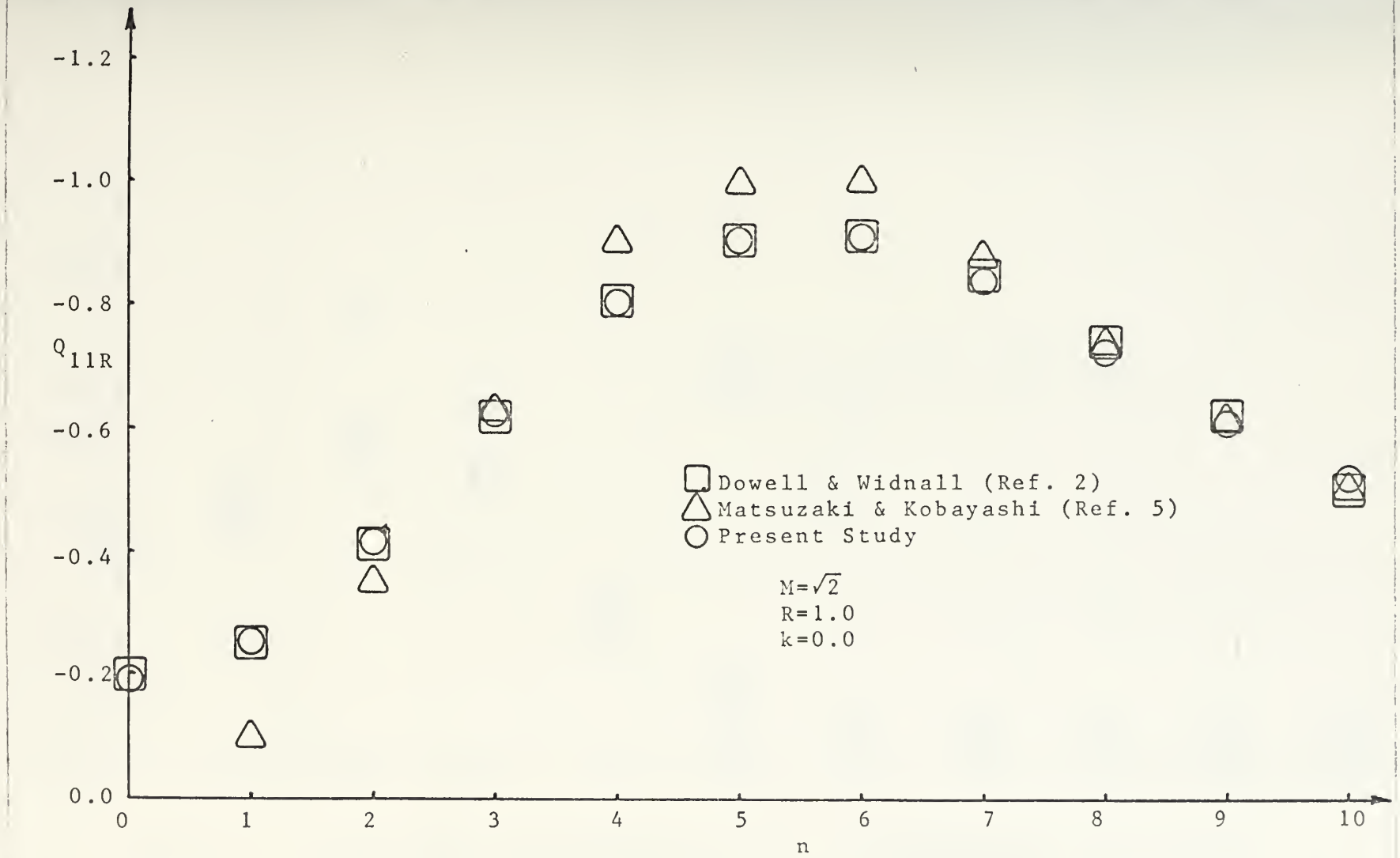


Figure 12. Generalized Aerodynamic Force vs. Circumferential Mode Number

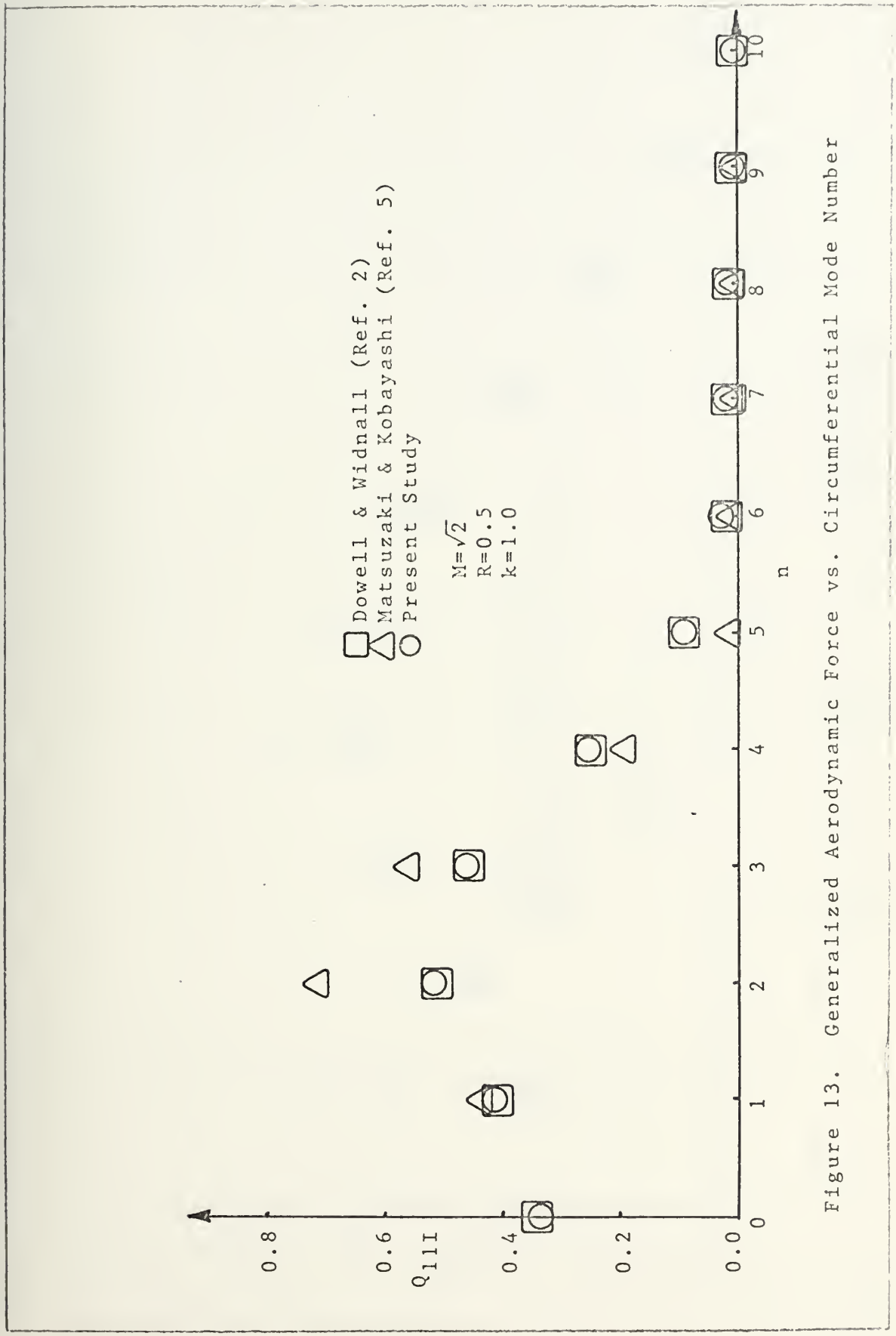


Figure 13. Generalized Aerodynamic Force vs. Circumferential Mode Number

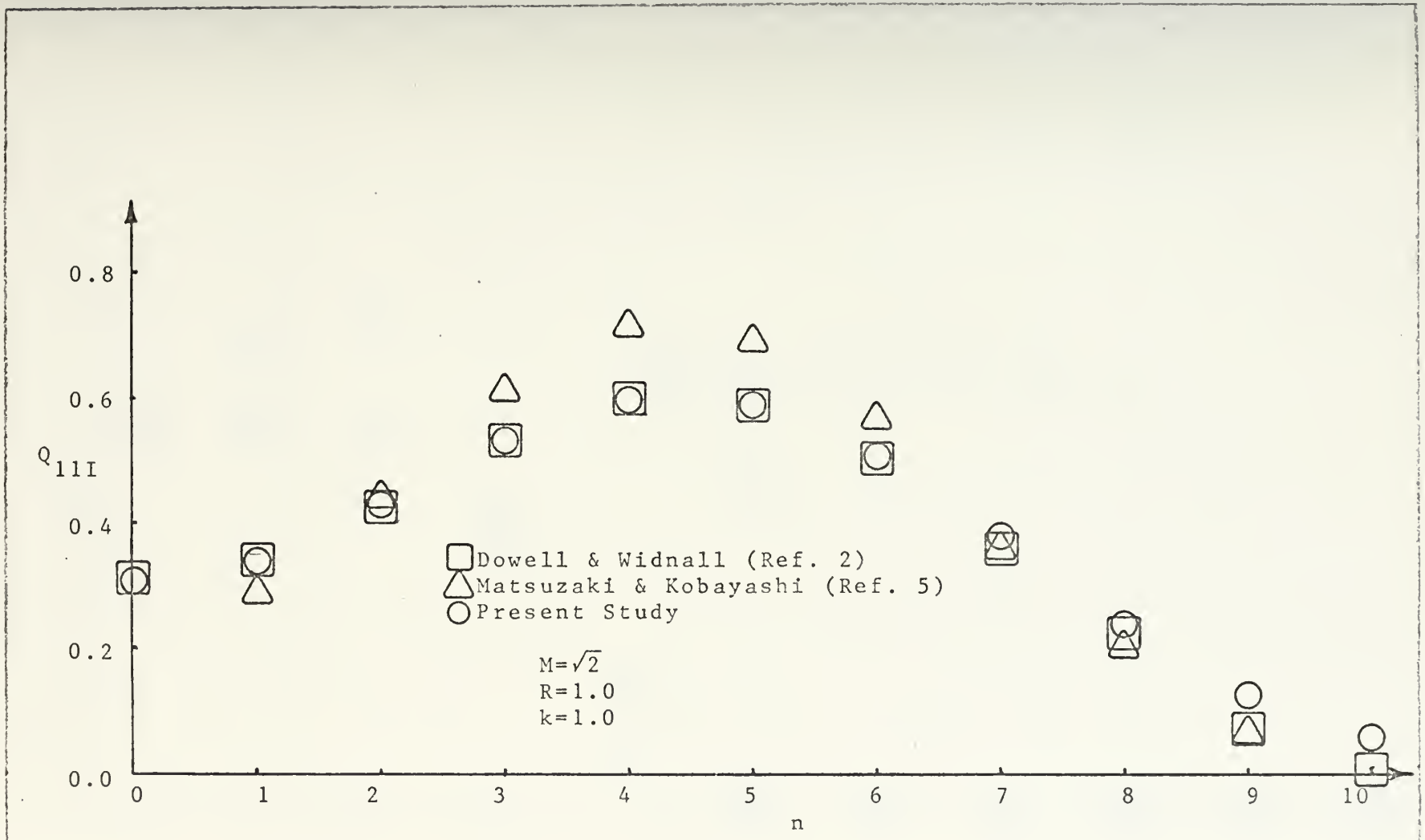


Figure 14. Generalized Aerodynamic Force vs. Circumferential Mode Number

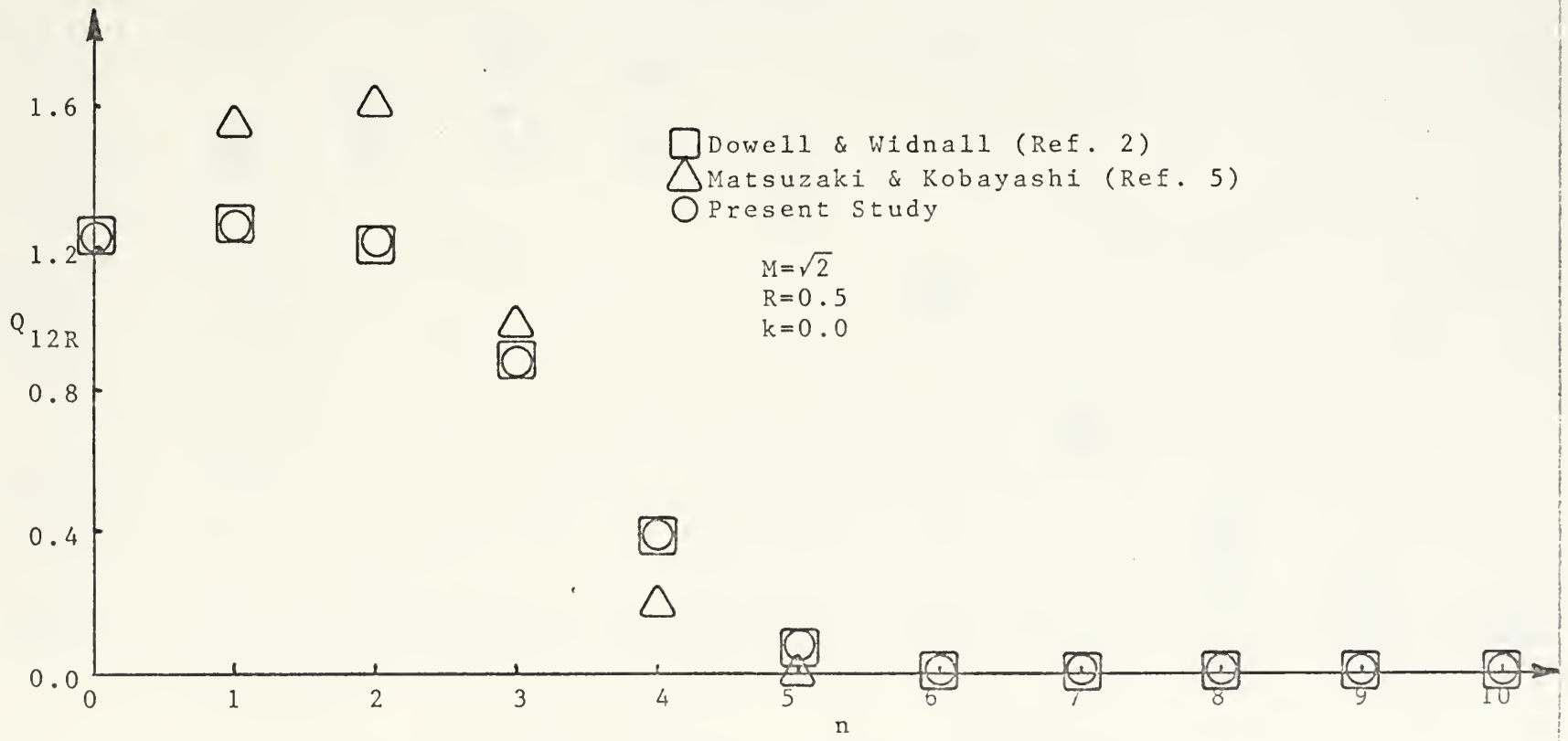


Figure 15. Generalized Aerodynamic Force vs. Circumferential Mode Number

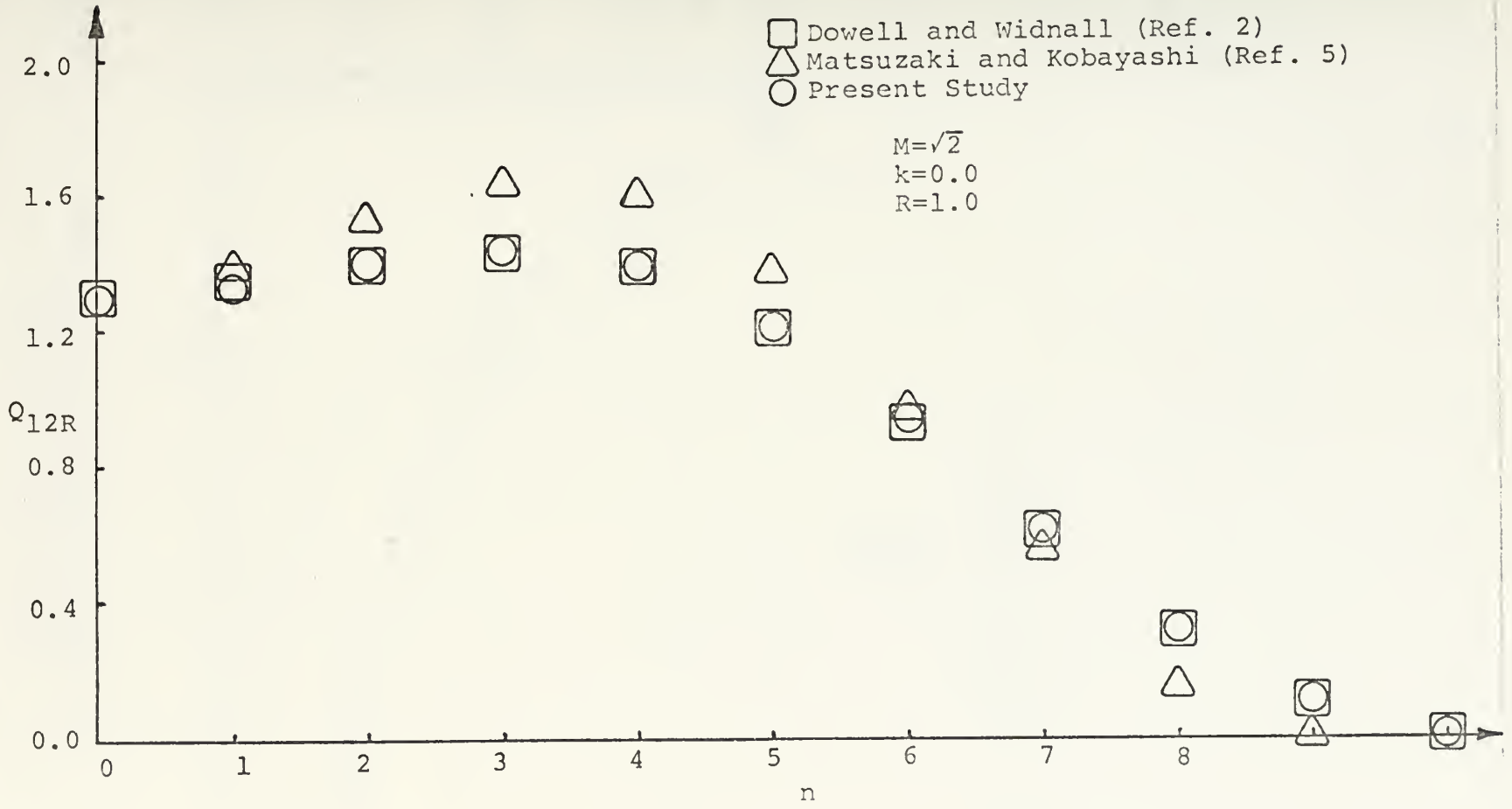


Figure 16. Generalized Aerodynamic Force vs. Circumferential Mode Number

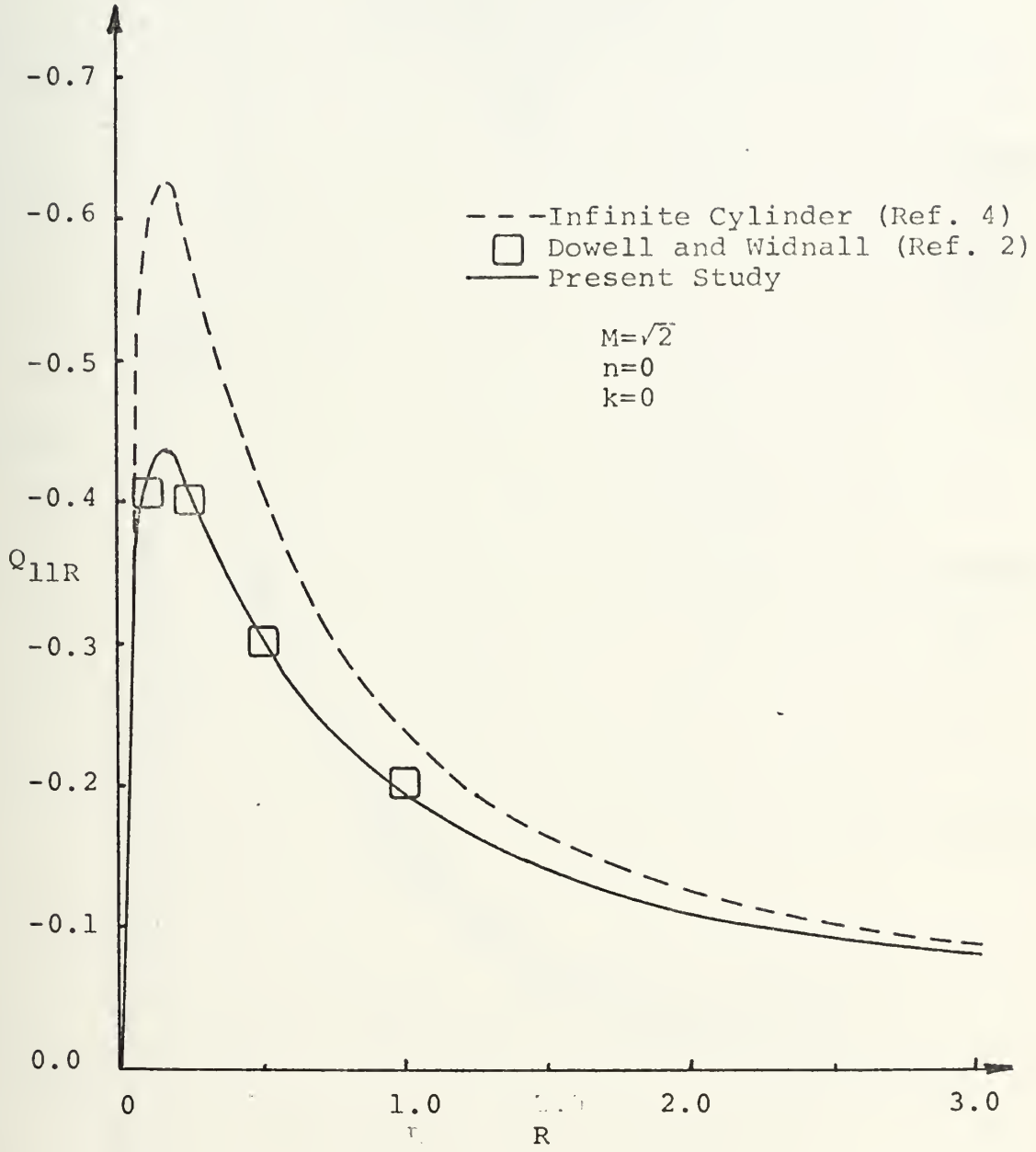


Figure 17. Generalized Aerodynamic Force vs. Radius to Length Ratio

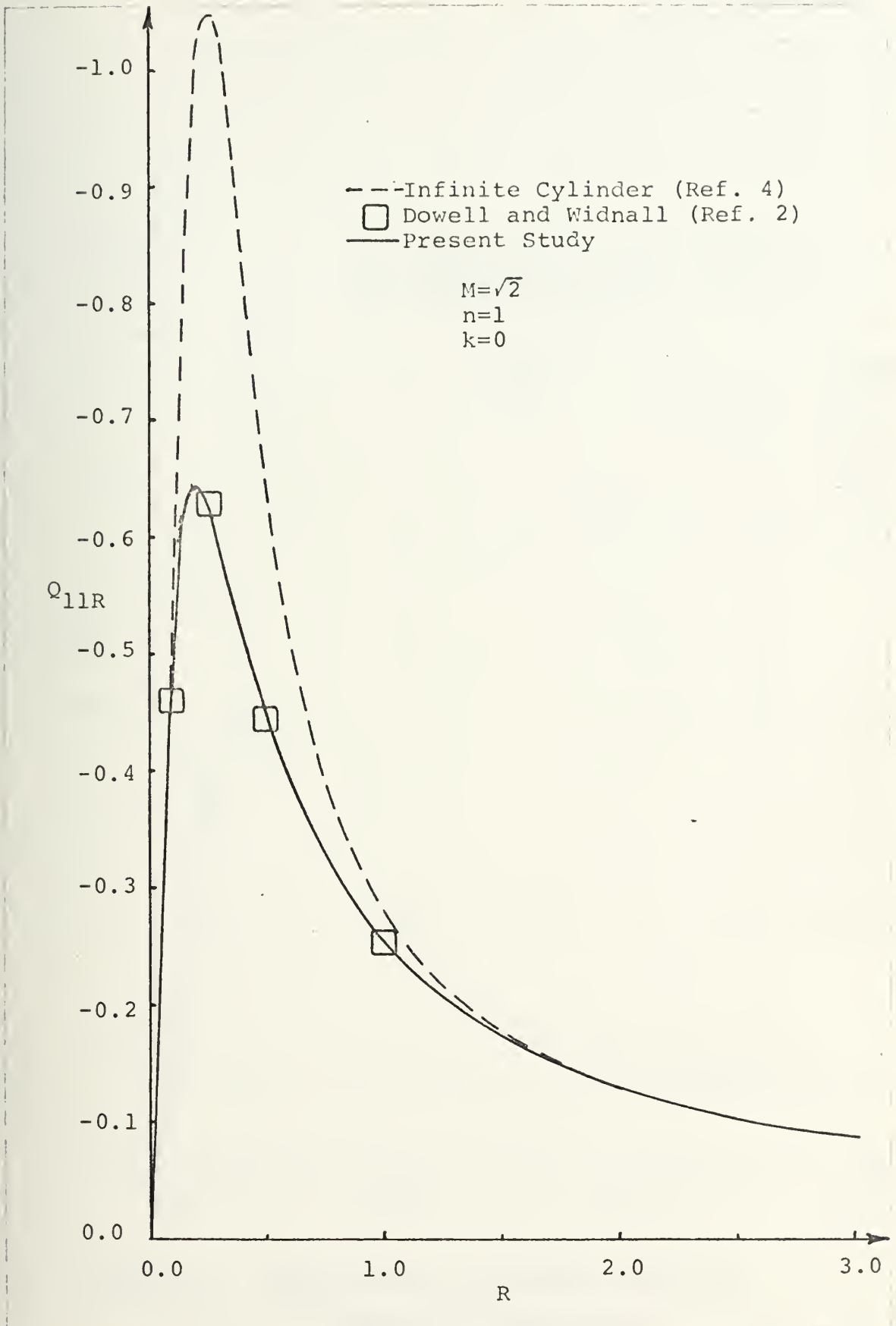


Figure 18. Generalized Aerodynamic Force vs. Radius to Length Ratio

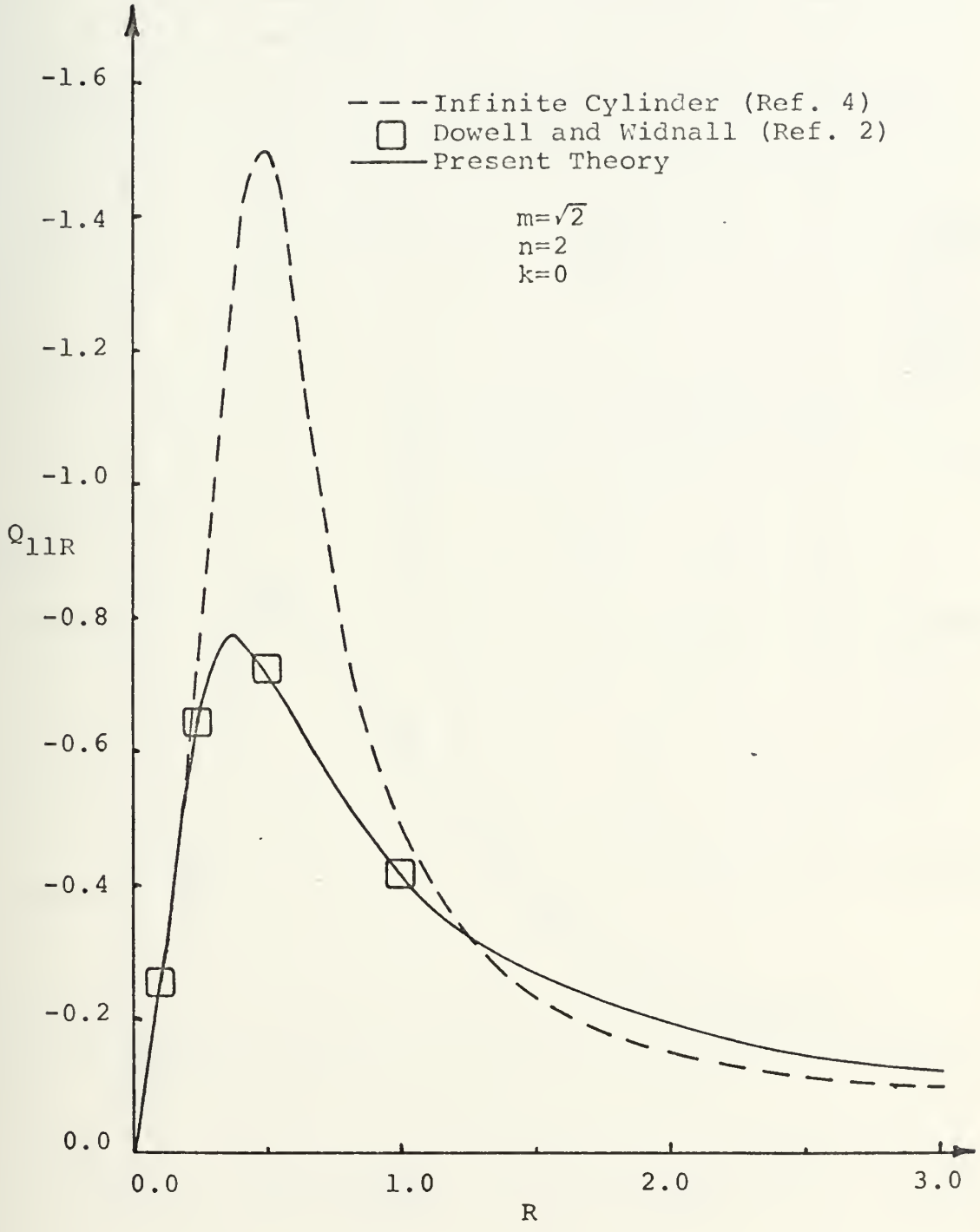


Figure 19. Generalized Aerodynamic Force vs. Radius to Length Ratio

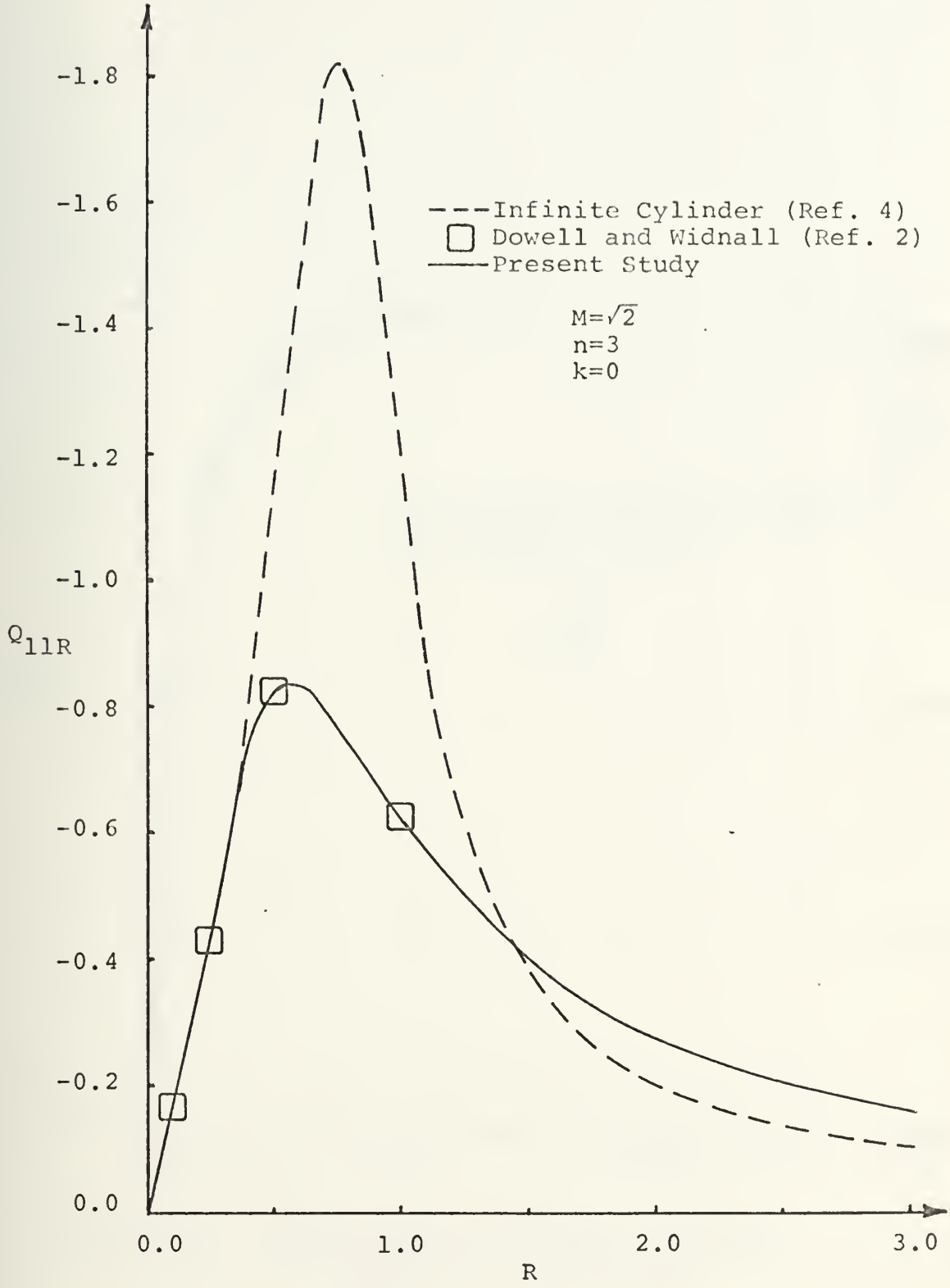


Figure 20. Generalized Aerodynamic Force vs. Radius to Length Ratio

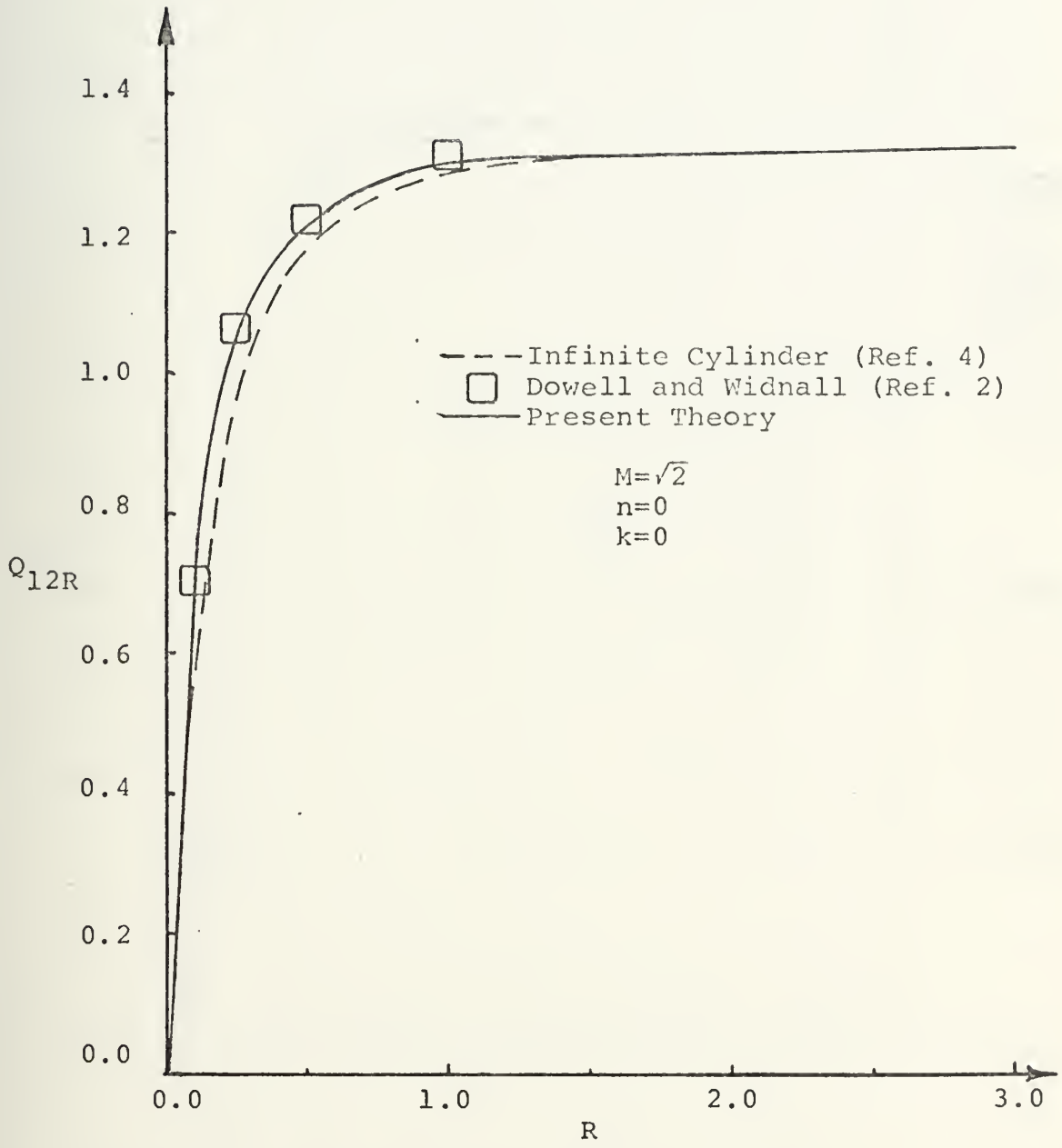


Figure 21. Generalized Aerodynamic Force vs. Radius to Length Ratio

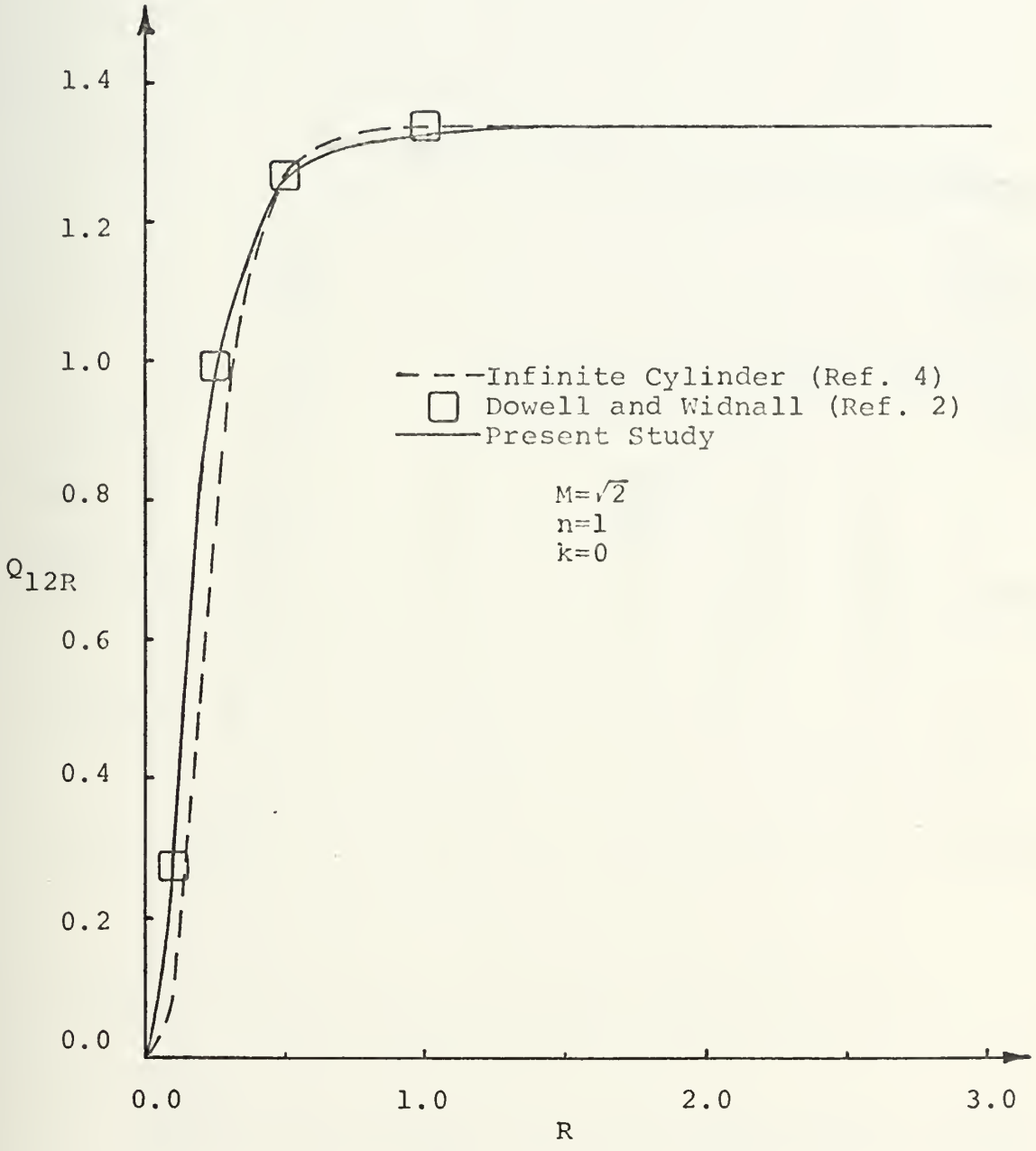


Figure 22. Generalized Aerodynamic Force vs. Radius to Length Ratio

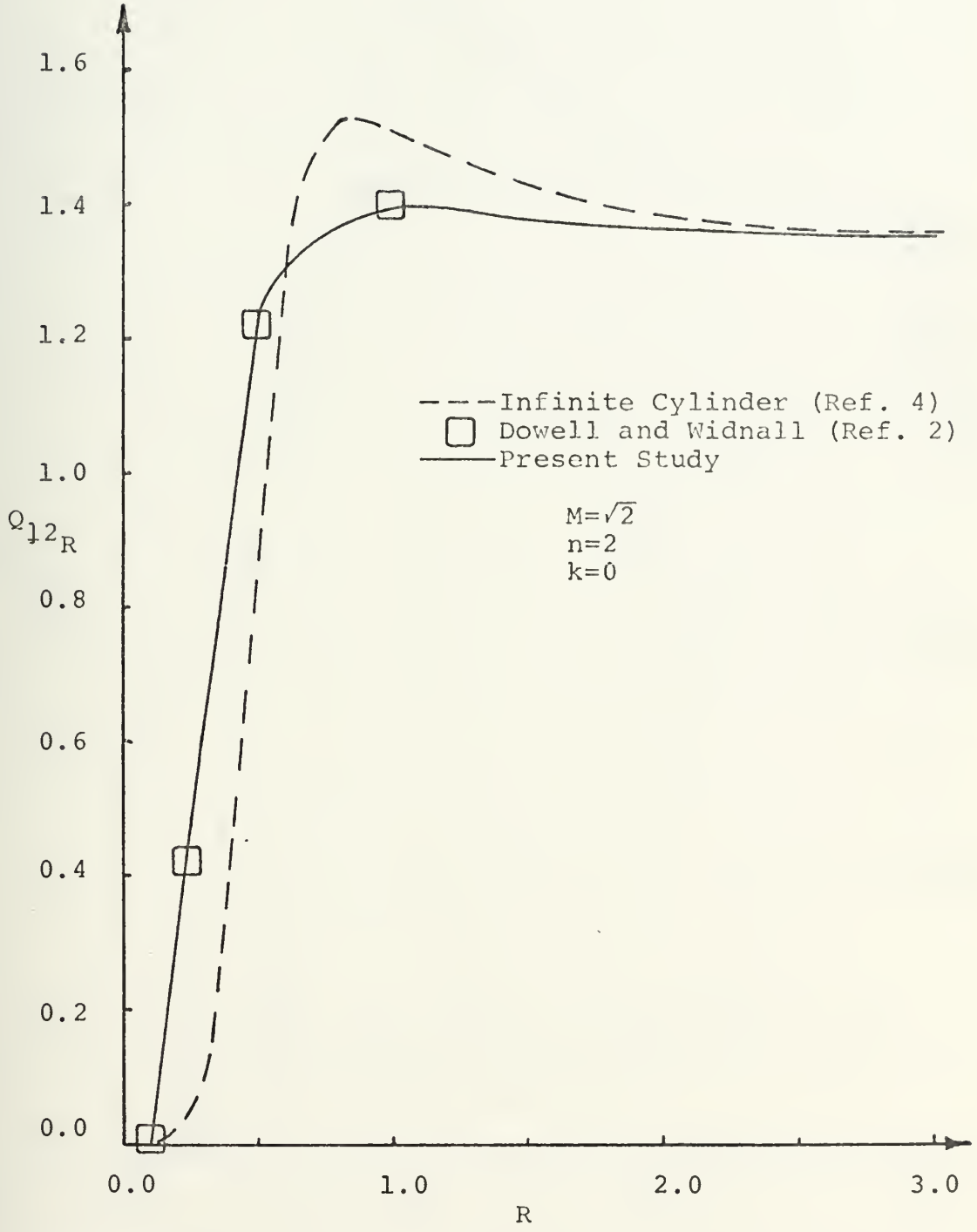


Figure 23. Generalized Aerodynamic Force vs. Radius to Length Ratio

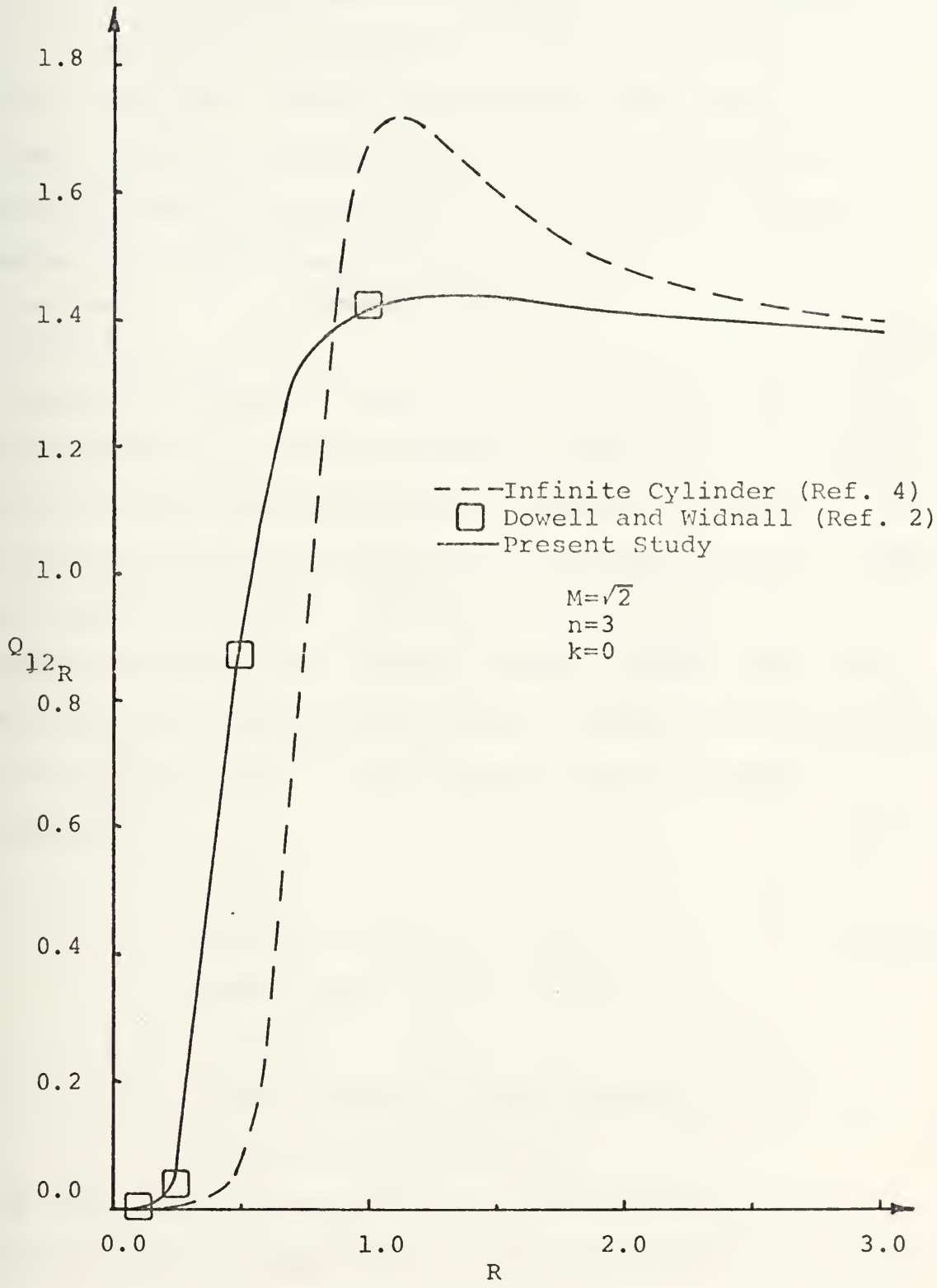


Figure 24. Generalized Aerodynamic Force vs. Radius to Length Ratio

Finally, Figure 25 shows the change in pressure distribution as a function of radius to length ratio for the case $M=\sqrt{2}$, m=1, n=0, k=0. These distributions were computed using the characteristics method outlined in Section III-A and the Sauer-Heinz theory as described in Section III-C. Complete agreement was found between the two methods.

B. CASCADE

Figures 26 through 29 show the instantaneous pressure distributions on the first and second blades of a finite cascade for four different values of reduced frequencies. The pressure difference between lower and upper blade surface is plotted for the instant when the airfoil is at its mean horizontal position and pitching nose up. Results for these cases were given very recently by L. E. Snyder, of Pratt and Whitney Aircraft (Ref. 8), for the following parameter combinations:

M = 1.5

Solidity = 1.33 (d = 0.317)

Stagger angle = 25° (β = 65°)

x_0 = 0.5

S = 40° (Blade 2 leads blade 1)

k = 0.04,0.332,0.832,1.25

It can be seen that appreciable differences exist in a number of cases, in particular in the wave reflection regions. These differences are not fully understood at this time.

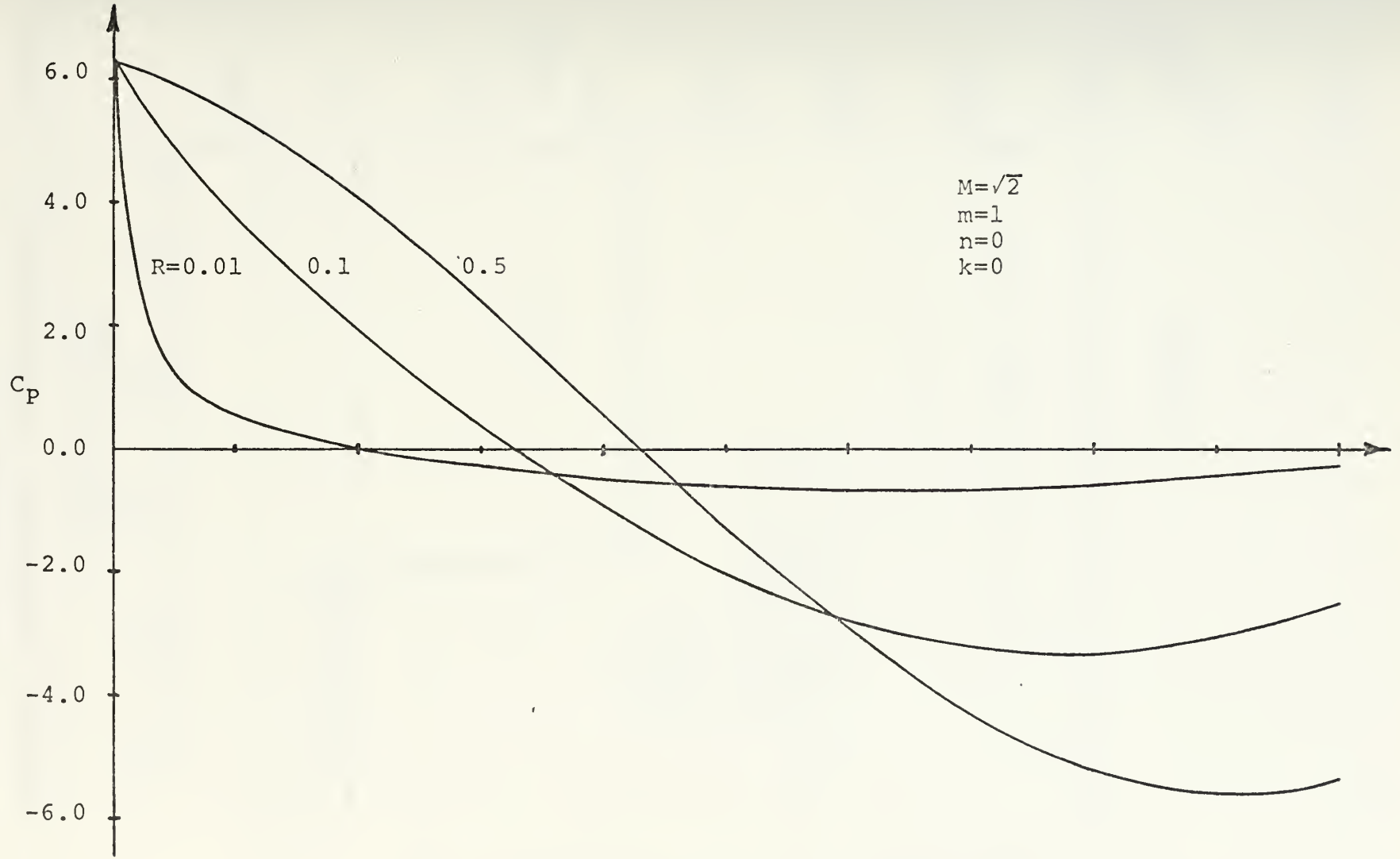


Figure 25. Pressure Distribution Over Wavy-Walled Cylinder

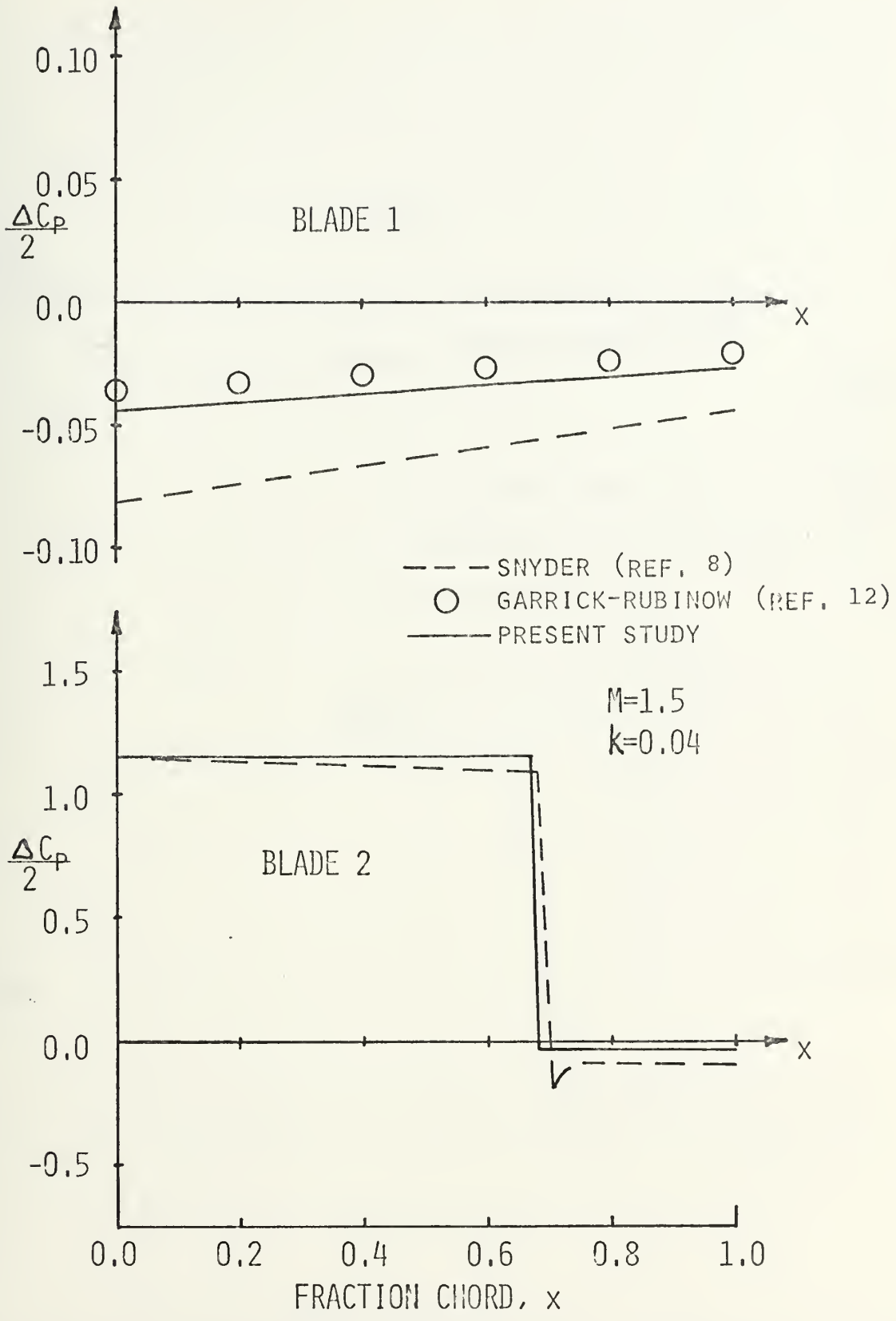


Figure 26. Cascade Pressure Coefficient Difference vs. Fraction Chord

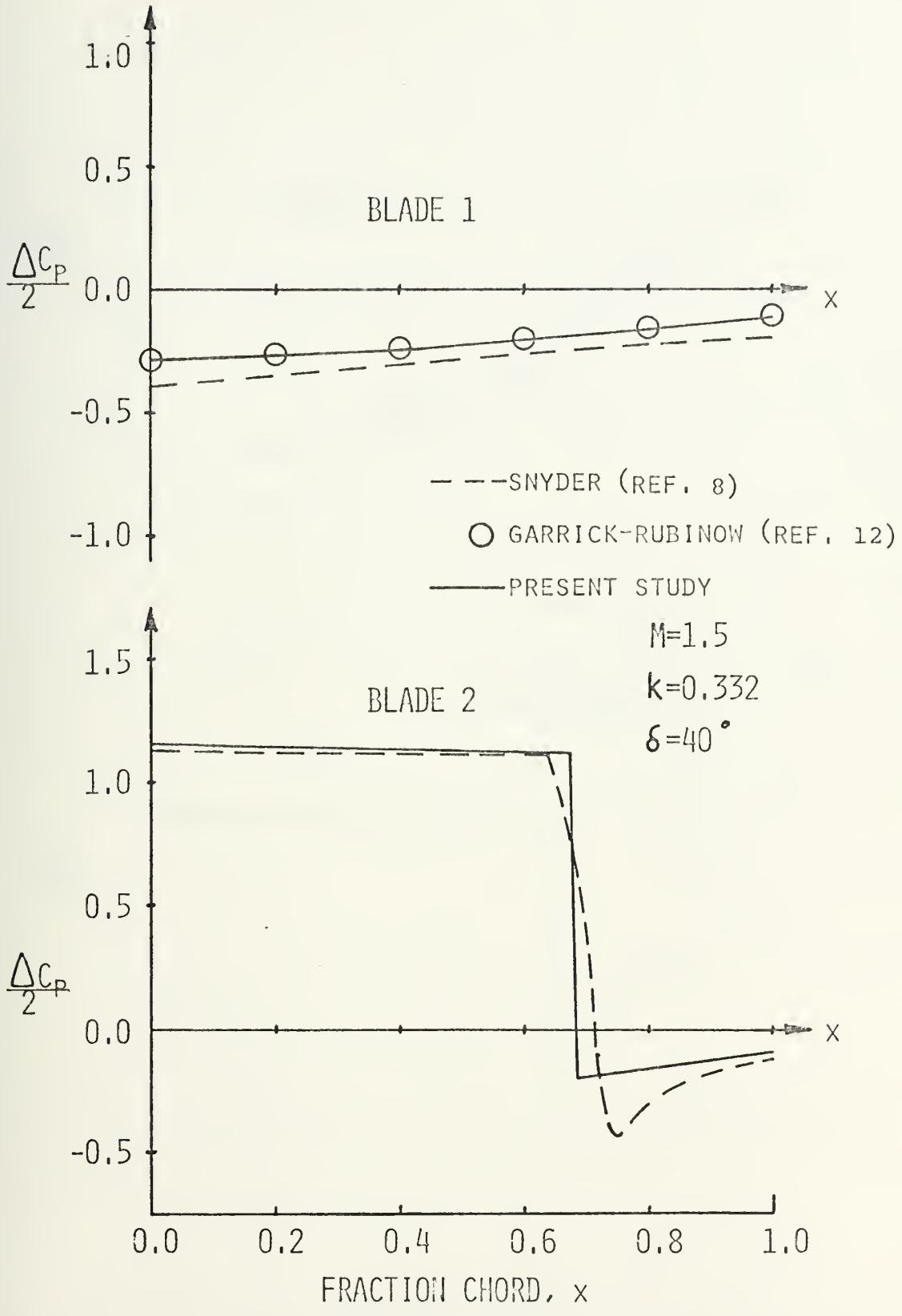


Figure 27. Cascade Pressure Coefficient Difference vs. Fraction Chord

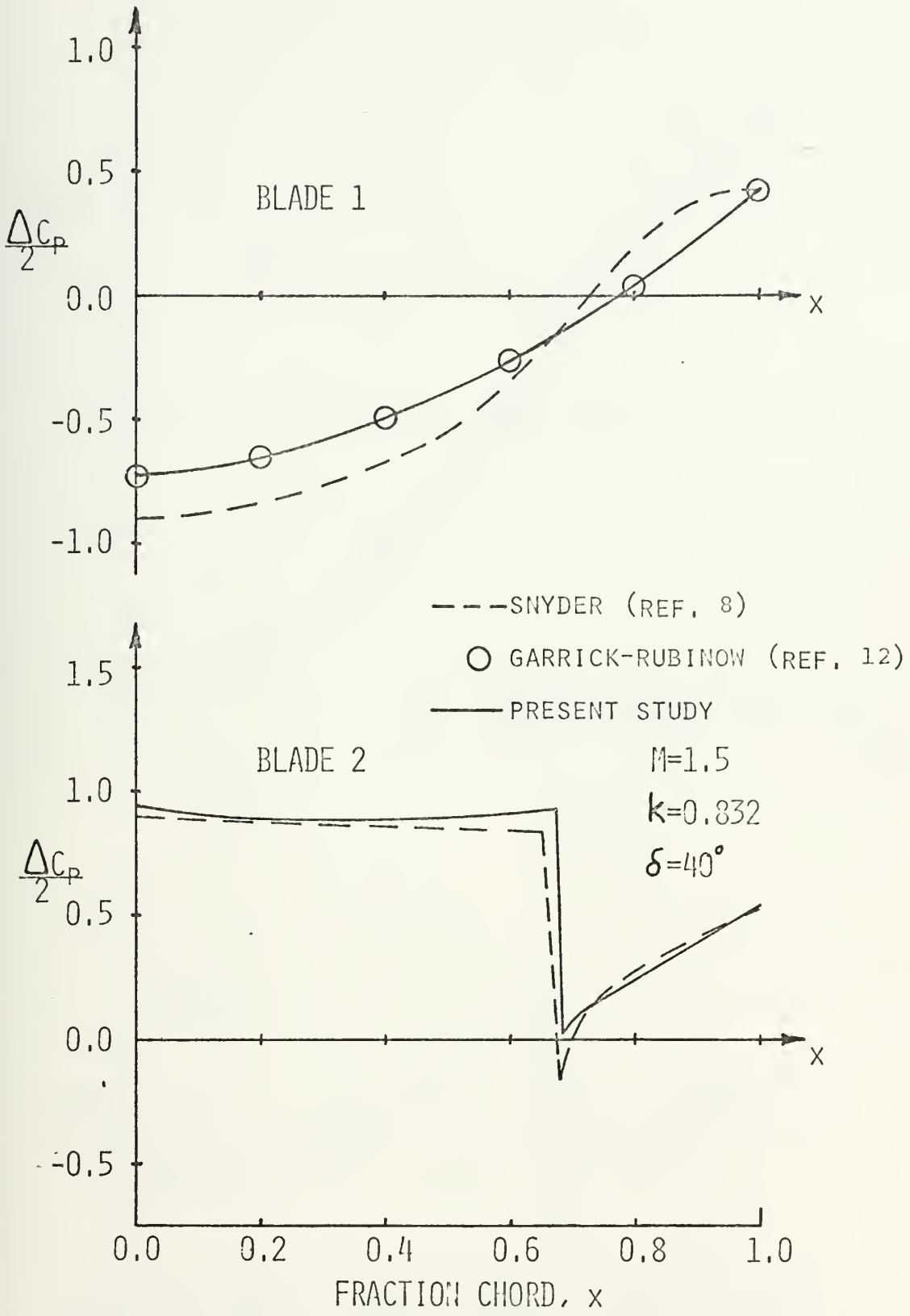


Figure 28. Cascade Pressure Coefficient Difference vs. Fraction Chord

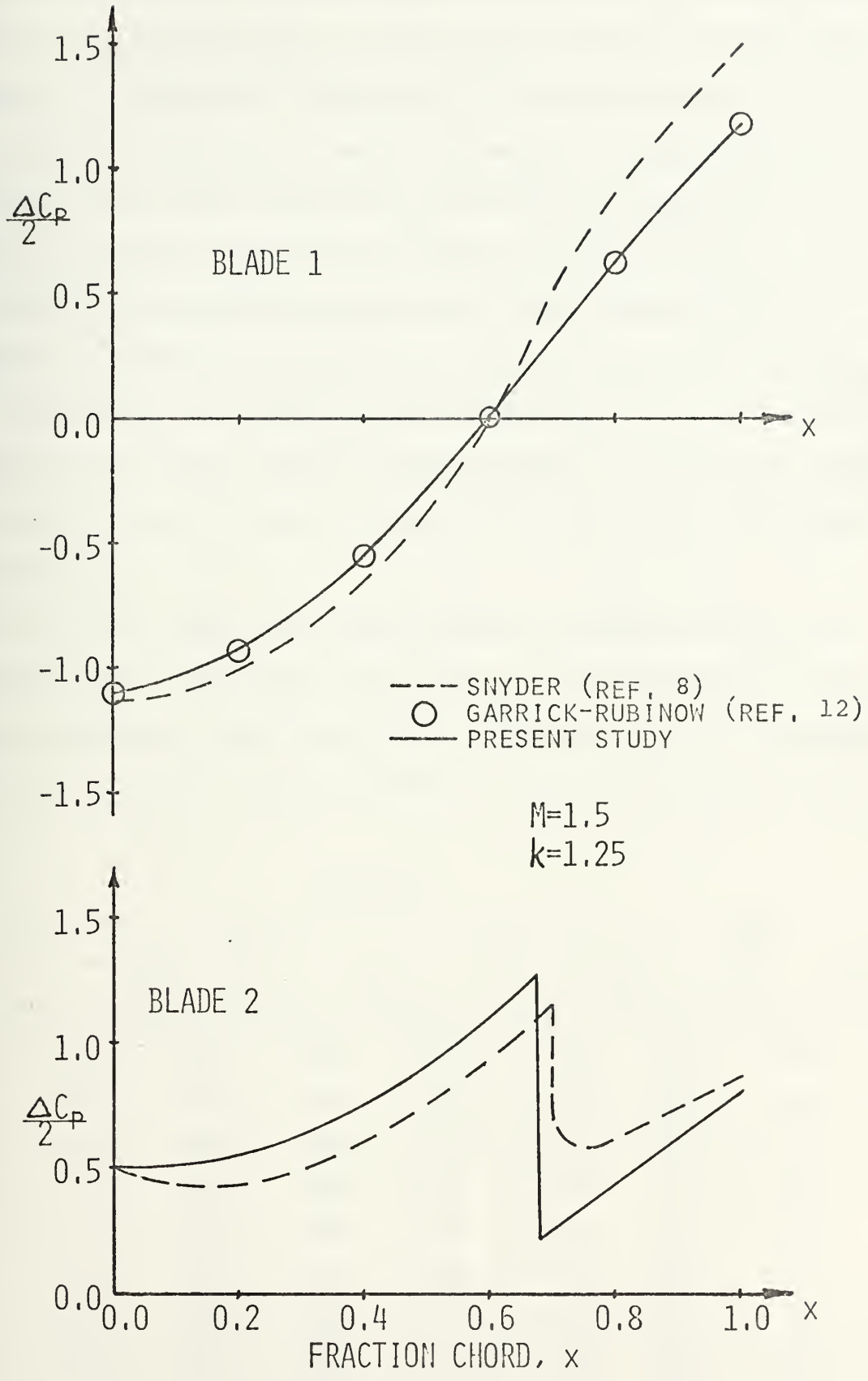


Figure 29. Cascade Pressure Coefficient Difference vs. Fraction Chord

Snyder uses a finite difference procedure and further work is required to evaluate the validity and accuracy of this procedure is compared to the method of characteristics procedure developed in the present work. However, a check could be made for the first blade solutions for which the method of singularities of Garrick and Rubinow (Ref. 12) is available. This solution was programmed according to the equations given in Appendix H and the results are shown in Figures 26 through 29. The method of characteristics solution is in complete agreement with the Garrick-Rubinow prediction whereas considerable deviations are seen to exist for the finite difference predictions.

A further check using the method of singularities was developed for two blades oscillating with zero phase difference. This approach is also described in Appendix H and a comparison with the method of characteristics is given in Table 1 below:

TABLE 1

k	0.1				0.5			
M	$\sqrt{2}$		1.8		$\sqrt{2}$		1.8	
x	M.O.C.	M.O.S.	M.O.C.	M.O.S.	M.O.C.	M.O.S.	M.O.C.	M.O.S.
0.1	.979	.979	.667	.660	.693	.694	.640	.486
0.2	.977	.977	.666	.661	.686	.687	.638	.511
0.3	.974	.974	.666	.662	.680	.681	.635	.536
0.4	.972	.972	.666	.663	.676	.677	.631	.562
0.5	.969	.969	.666	.664	.674	.675	.627	.588
0.6	.966	.966	.665	.666	.675	.676	.623	.614
0.7	.963	.963	.665	.667	.678	.679	.618	.641
0.8	.959	.959	.665	.668	.682	.683	.614	.668
0.9	.956	.956	.664	.669	.689	.690	.607	.694

Further method of characteristics results for a finite subsonic leading edge cascade are shown in Figures 30 through 32 for the following parameter combinations:

$M=\sqrt{2}$ $\quad x_0=0.5$

$d=1.0$ $\quad k=0.5$

$\beta=59.8°$ $\quad \delta=120°$

As mentioned previously, the computer program is written sufficiently general so that the pressure distribution can be computed and printed out for arbitrary specified Mach number, reduced frequency, stagger angle, blade spacing, and number of blades subject to the following constraint:

$$\nu \tan \alpha \left[(N-1)\ (\tan \beta - \cot \alpha) + \frac{1}{d}\right] < 800 \qquad (227)$$

Equation (227) is the expression which governs the maximum grid fineness that may be used with a given set of input parameters. This constraint is necessary because the program is dimensioned such that the maximum number of grid points allowed on any one right-running Mach line is 400.

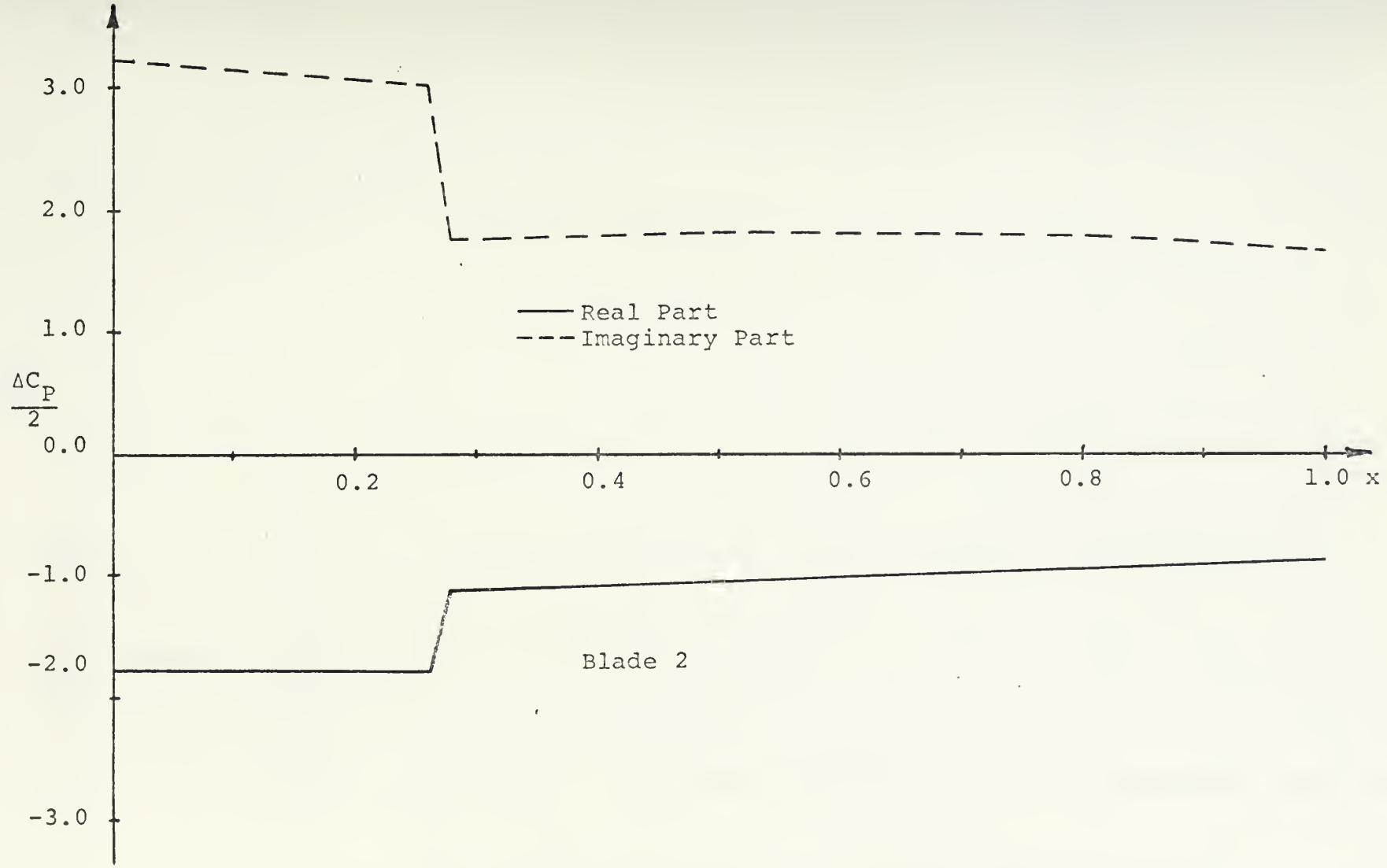


Figure 30. Pressure Coefficient Difference vs. Fraction Chord

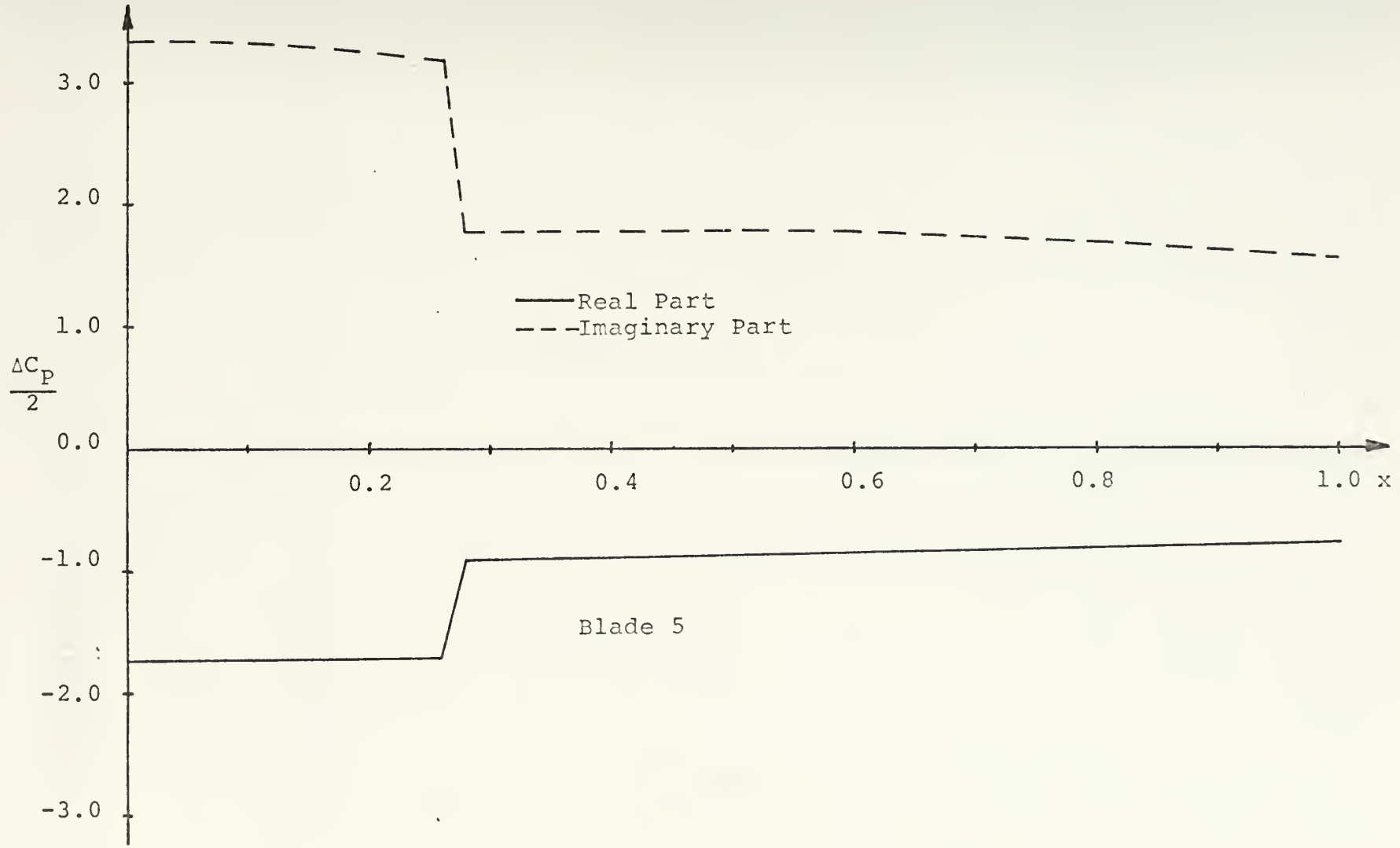


Figure 31. Pressure Coefficient Difference vs. Fraction Chord

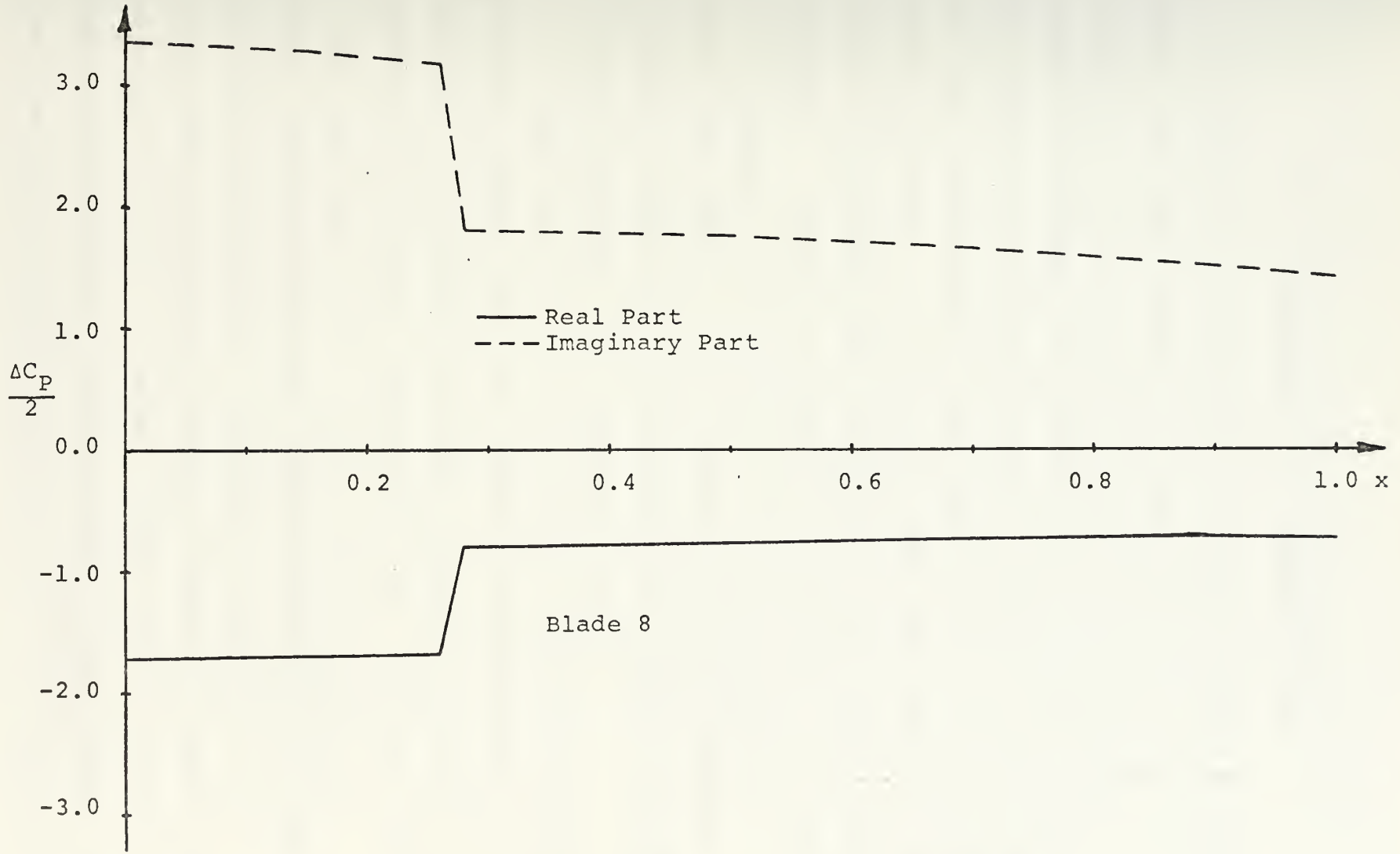


Figure 32. Pressure Coefficient Difference vs. Fraction Chord

VI. CONCLUSIONS AND RECOMMENDATIONS

A solution of the complete linearized unsteady potential equation has been developed for supersonic flow past vibrating cylindrical shells using the method of characteristics. Pressure distributions and generalized aerodynamic forces were computed for shells of arbitrary radius-to-length ratios, axial and circumferential wave numbers, Mach number, and reduced frequencies. A solution for this same problem was previously obtained by Dowell and Widnall using Laplace transform techniques. Excellent agreement was obtained in all cases. It is believed that the present method can be applied more easily than methods requiring complex variable techniques. It should be noted that the procedure can easily be modified to include arbitrary axial deflection modes.

A further solution of the complete linearized unsteady potential equation using the method of characteristics was developed for supersonic flow past vibrating flat plate cascades with subsonic leading edge locus. Pressure distributions were computed for finite cascades of arbitrary stagger and interblade phase angles performing small amplitude pitch oscillations. Preliminary comparisons with a recent finite difference solution by Verdon and Snyder showed reasonable agreement. However, further work is necessary to assess the validity and accuracy of the present method. This should be done by further comparisons with the Verdon and

Snyder procedure and Fourier and Laplace transform methods currently under development at Stevens Institute of Technology, General Motors Corporation (Detroit Diesel, Allison Division), and General Electric. Also, flutter calculations should be carried out using the method of characteristics predictions of the oscillatory aerodynamic forces which would allow comparisons with the Pratt and Whitney cascade flutter tests.

APPENDIX A

CYLINDRICAL SHELL COMPUTER PROGRAM VARIABLES

A. LOGIC VARIABLES

The following variables appear in the logic statements of the program:

IHAVEP	The number of the computation along any right-running Mach line. IHAVEP is used as the first subscript for a grid point.
ISWTCH	Switch variable used to designate the right-running Mach line being calculated, alternately set equal to "1" or "2" and used as a second subscript.
JSWTCH	Switch variable used to designate the previously computed right-running Mach line, always the opposite of ISWTCH and used as a second subscript.
KOUNT	The value of IHAVEP at the cylinder surface.
ML	The number of grid points along the initial Mach line and along the cylinder surface.

B. QUANTITY VARIABLES

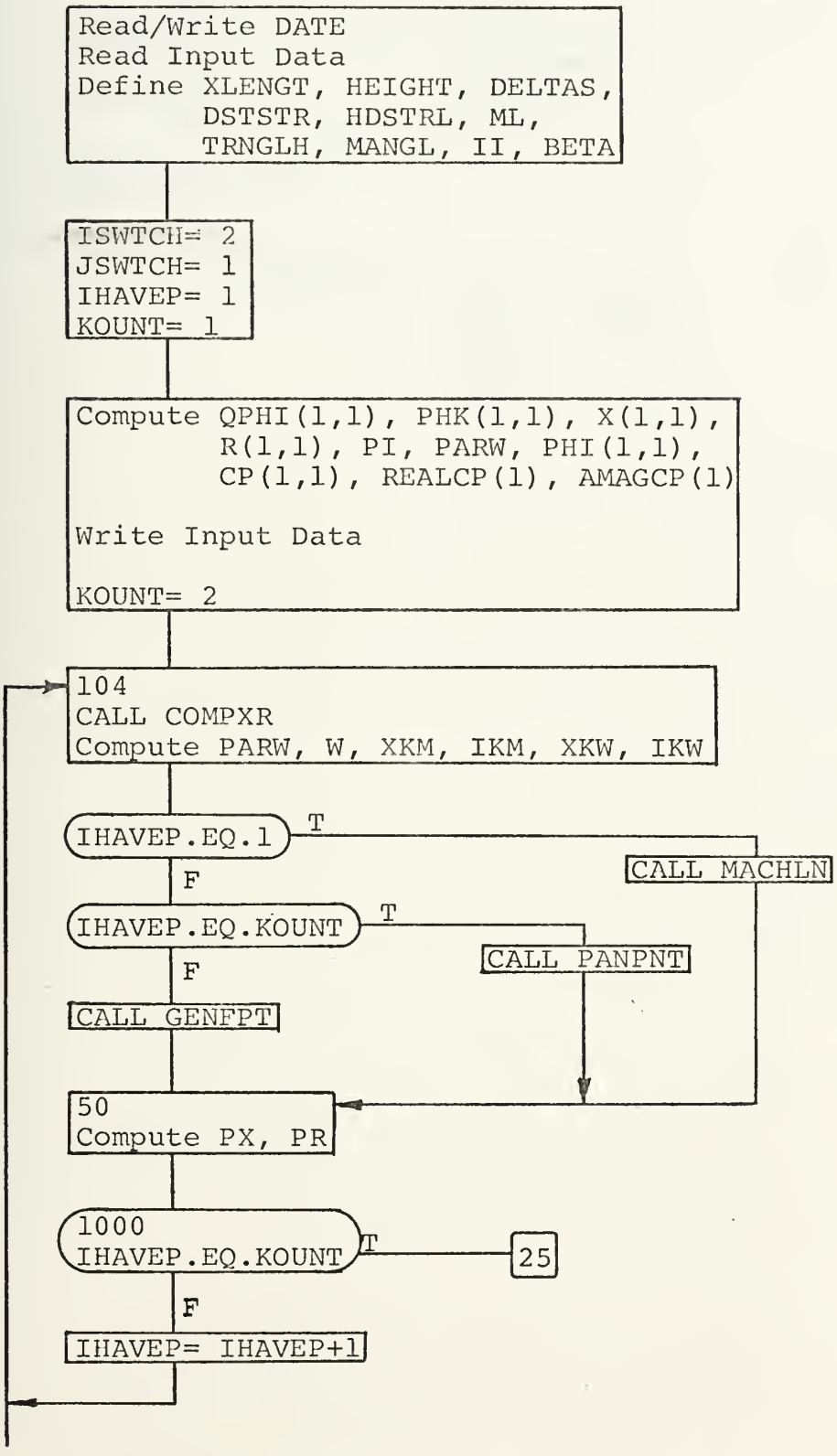
The following variables in the program take on the values defined below. All are floating point quantities unless otherwise stated. The dimensioned variables are listed first.

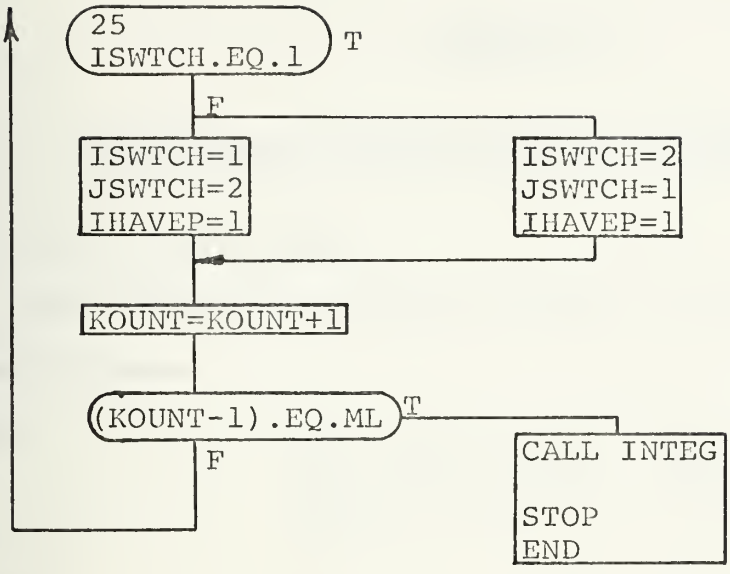
AMAGCP(400)	The imaginary part of the pressure coefficient.
CP(400)	The complex pressure coefficient.
PHI(400,2)	The derivative of the velocity potential along the right-running Mach line.
PHK(400,2)	The derivative of the velocity potential along the left-running Mach line.
QPHI(400,2)	The velocity potential.

REALCP(400,2)	The real part of the pressure coefficient.
X,R(400,2)	The x and r coordinates of a grid point.
A,B,C,D,E,F	See equations (57) - (62)
BETA	The square root of the quantity Mach number squared minus one.
DELTAS	The step size along the characteristics.
DSTSTR	The horizontal distance between grid points.
FINGRD	The grid fineness ratio. An even integer less than 400.
HDSTRL	One-half DSTSTR.
HEIGHT	The increment HDSTRL $\frac{1}{2}$ tan α.
II	The square root of minus one.
IKM	The factor, ik $\frac{M}{M^2-1}$
IKW	The quantity, ik sin $m\pi x$
IPHI1, IPHI2	The velocity potential as described in equations (51) and (54).
K	The reduced frequency.
MACH	The free-stream Mach number
MANGL	The Mach angle, α.
MFREQ	The axial mode number. An integer.
NFREQ	The circumferential mode number. An integer.
NR	r, as defined in equation (68). An integer.
PARW	The partial derivative of the deflection taken with respect to x.
PI	3.141593
PX	The partial derivative of the velocity potential taken with respect to x.
RADIUS	The radius to length ratio.
W	The cylinder deflection.

APPENDIX B

CYLINDRICAL SHELL FLOW DIAGRAM




25
ISWTCH.EQ.1
T
F
ISWTCH=1
JSWTCH=2
IHAVEP=1
ISWTCH=2
JSWTCH=1
IHAVEP=1
KOUNT=KOUNT+1
(KOUNT-1).EQ.ML
T
F
CALL INTEG
STOP
END

APPENDIX C

OSCILLATING CASCADE COMPUTER PROGRAM VARIABLES

A. LOGICAL VARIABLES

The following variables appear in the logic statements of the program:

ICO	Set equal to "1" when a blade or wake computation is made. Set equal to "0" after the next general flow field computation occurs.
IHAVEP	The number of the computation along any right-running Mach line starting with zero at the initial left-running Mach line. The value IHAVEP+1 is the first subscript for a flow quantity.
ISWTCH	Switch variable used to designate the right-running Mach line being calculated, alternately set equal to "1" or "2" and used as a second subscript.
JLINE	The Mach line counter incremented each time a right-running Mach line is completed. Reset equal to the value of IHAVEP at the leading edge of a new blade.
JSWTCH	Switch variable used to designate the previously computed right-running Mach line, always the opposite of ISWTCH and used as a second subscript.
LCOUNT	The value of IHAVEP at a surface or wake grid point.
LIMIT	The maximum value of IHAVEP.
LIMW	The value of LCOUNT at the trailing edge of a blade.
MAXI	The value of JLINE on a right-running Mach line which intersects a blade's leading edge.
NEWDST	The number of Δx increments in a blade chord.

NSTPTS — The number of Δx increments between the initial Mach line and the leading edge of the first blade.

NUM — Set equal to one less than the number of the blade encountered on a Mach line.

NUMBLD — The number of blades in the cascade.

NUMOLD — Preserves the maximum value of NUM.

NUMPI — NUMOLD plus one.

OLDL — Preserves the value of LCOUNT at the highest blade or wake yet encountered.

B. QUANTITY VARIABLES

The following variables in the program take on the values defined below. All are floating point quantities unless otherwise stated. The dimensioned variables are listed first.

A(8,8) — Matrix of coefficients in wake computation.

B(8) — Right-hand side of equations (194) - (201)

U22R,U22I,V22R,V22I,C22R,C22I(400,2). The velocities, real and imaginary of general flow field, upper blade, and upper wake points. Equivalent to the respective Teipel amplitude functions, U_{22R} through C_{22I}.

U33R,U33I,V33R,V33I,C33R,C33I(400,2). Same as above, but used only at lower blade and lower wake points.

X,Y(400,2) — The x and y coordinates based on chord length.

AI — A_I as defined in equation (122).

ARG — The argument $k \frac{M^2}{M^2-1} x$

BI — B_I as defined in equation (123).

DELTA — The interblade phase angle in radians.

DELTAS — Incremental distance between grid points along Mach line.

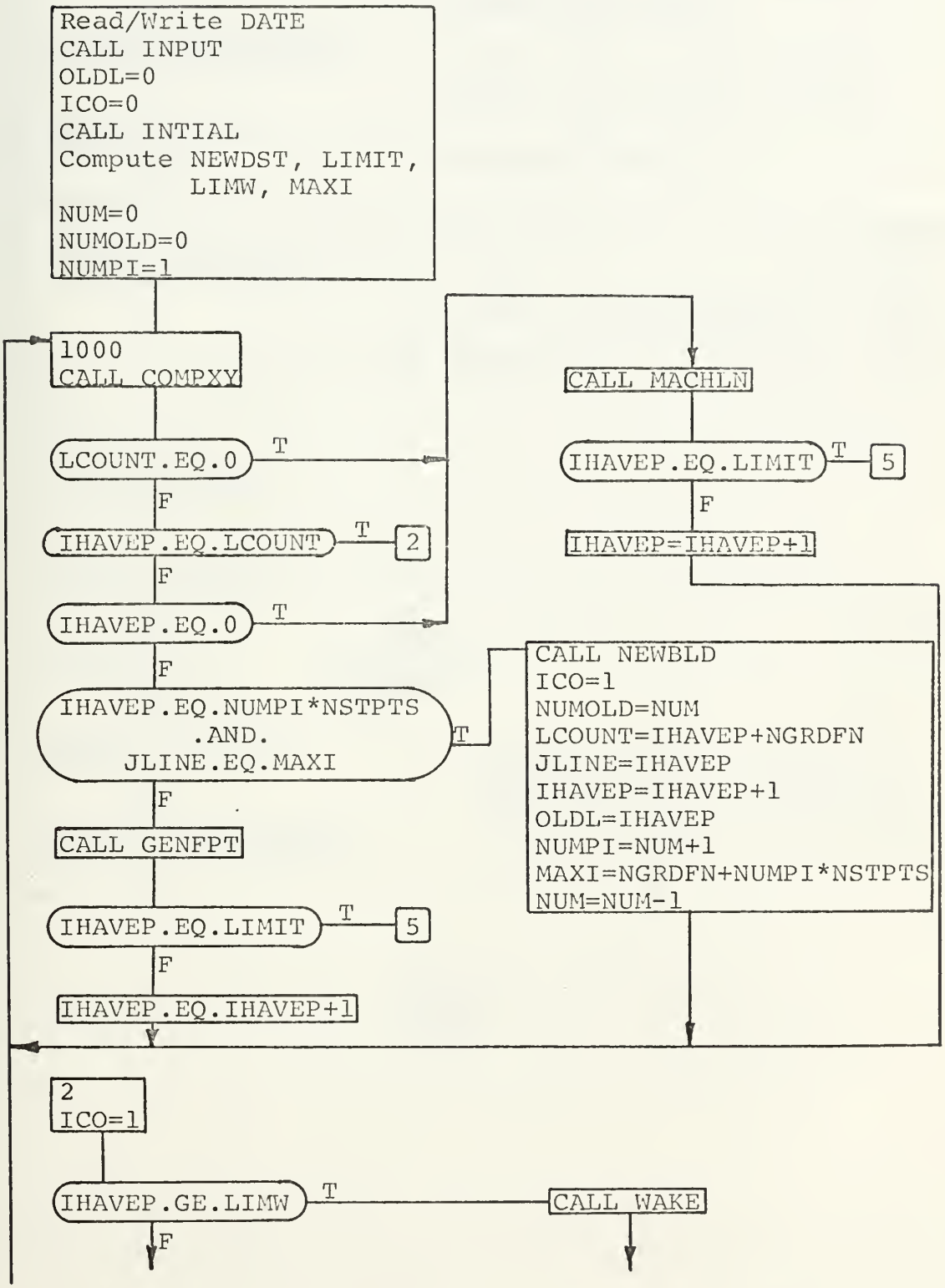
DSTSTR — The increment Δx.

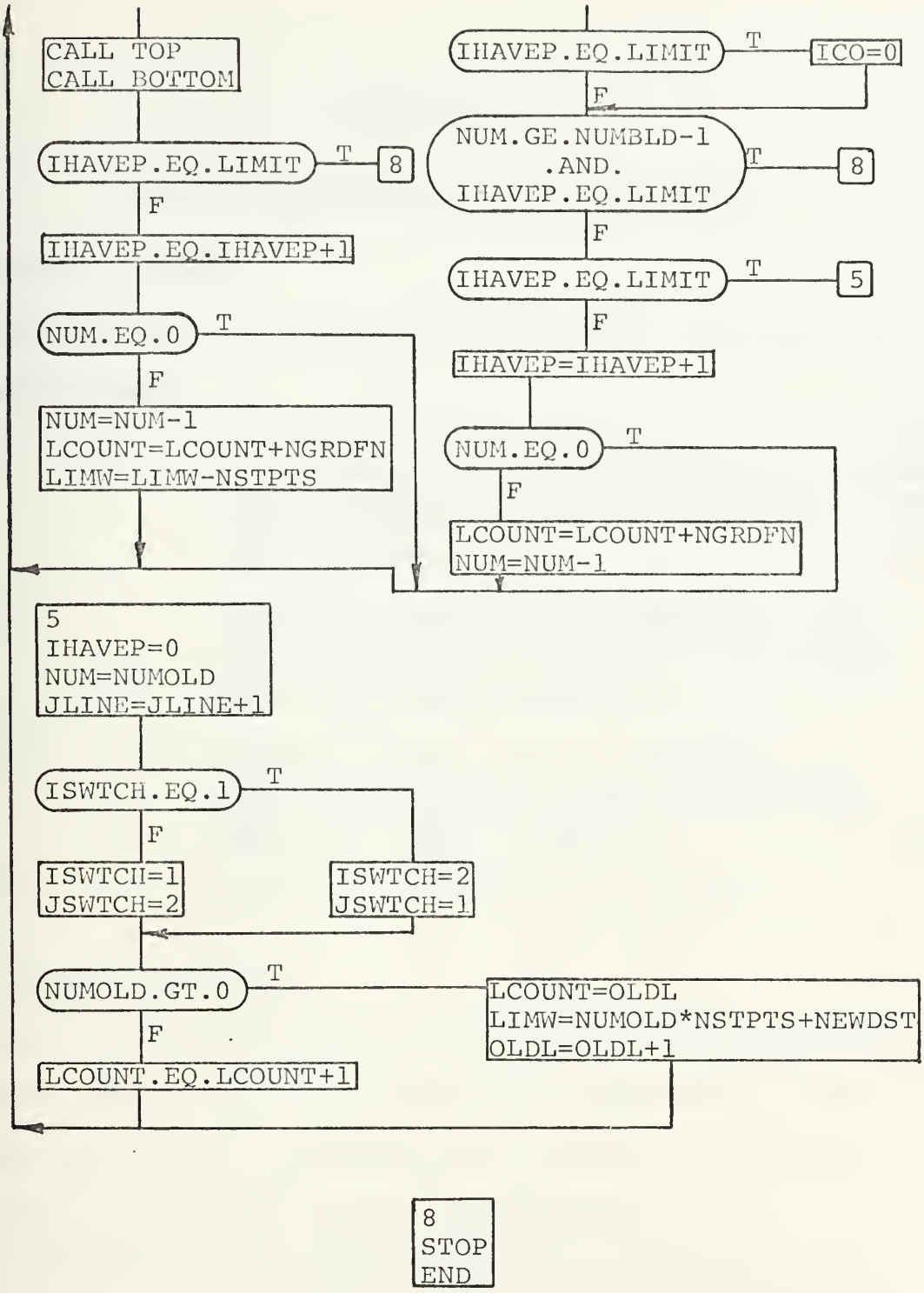
FAZE	The interblade phase angle in degrees.
FMANGL	Mach angle, α.
FNGDPT	Floating point NGRDFN.
FSTRMN	The free-stream Mach number.
G1,...,G9	Intermediate factors used in velocity computations in GENFPT, TOP, BOTTOM, and NEWBLD.
HDSTRL	The increment $\Delta x/2$.
K12R,...,K56I	$K_{12R},\ldots,K_{56I}$ as defined in equations (116) - (121).
KS	Dummy variable used in SIMQ. An integer.
NGRDFN	The grid fineness ratio. An even integer, less than 400.
REDFRQ	The reduced frequency.
STAG	The distance the second blade is staggered back from the y-axis.
STGR	The distance from the initial Mach line to the second blade's leading edge.
STGANG	The stagger angle in degrees.
S	The factor $\sqrt{M^2-1}$.
TNWDST	The vertical distance between adjacent blades.
TRNGLH	The increment $\frac{1}{2}\Delta x \cos \alpha$.
U	The factor $\cos(k \frac{M^2}{M^2-1} x)$.
V	The factor $\sin(k \frac{M^2}{M^2-1} x)$.
VIPANL	The imaginary part of the normal flow velocity at the first blade's leading edge.
VRPANL	The real part of the normal flow velocity at the first blade's leading edge.
W	The factor $\frac{M^2}{\sqrt{M^2-1}}$

XSUBO	The elastic axis position based on chord length.
XLNGTH	The length of the initial Mach line between the planes of adjacent blades.
XN	Floating point NUM.
XNEW	The x-coordinate on a blade based on the blade's leading edge.

APPENDIX D

CASCADE FLOW DIAGRAM




CALL TOP
CALL BOTTOM
IHAVEP.EQ.LIMIT
T
8
F
IHAVEP.EQ.IHAVEP+1
NUM.EQ.0
T
F
NUM=NUM-1
LCOUNT=LCOUNT+NGRDFN
LIMW=LIMW-NSTPTS
IHAVEP.EQ.LIMIT
T
ICO=0
F
NUM.GE.NUMBLD-1
.AND.
IHAVEP.EQ.LIMIT
T
8
F
IHAVEP.EQ.LIMIT
T
5
F
IHAVEP=IHAVEP+1
NUM.EQ.0
T
F
LCOUNT=LCOUNT+NGRDFN
NUM=NUM-1
5
IHAVEP=0
NUM=NUMOLD
JLINE=JLINE+1
ISWTCH.EQ.1
T
F
ISWTCH=1
JSWTCH=2
ISWTCH=2
JSWTCH=1
NUMOLD.GT.0
T
F
LCOUNT=OLDL
LIMW=NUMOLD*NSTPTS+NEWDST
OLDL=OLDL+1
LCOUNT.EQ.LCOUNT+1
8
STOP
END

APPENDIX E

SAUER-HEINZ COMPUTER PROGRAM VARIABLES

A. LOGIC VARIABLES

The following variables appear in the logic statements of the program:

IHAVEP	The number of the computation along any right-running Mach line starting with zero at the initial left-running Mach line. The value IHAVEP+1 is the first subscript for a flow quantity.
ISWTCH	Switch variable used to designate the right-running Mach line being calculated, alternately set equal to "1" or "2" and used as a second subscript.
JSWTCH	Switch variable used to designate the previously computed right-running Mach line, always the opposite of ISWTCH and used as a second subscript.
KOUNT	The value of IHAVEP at the surface.

B. QUANTITY VARIABLES

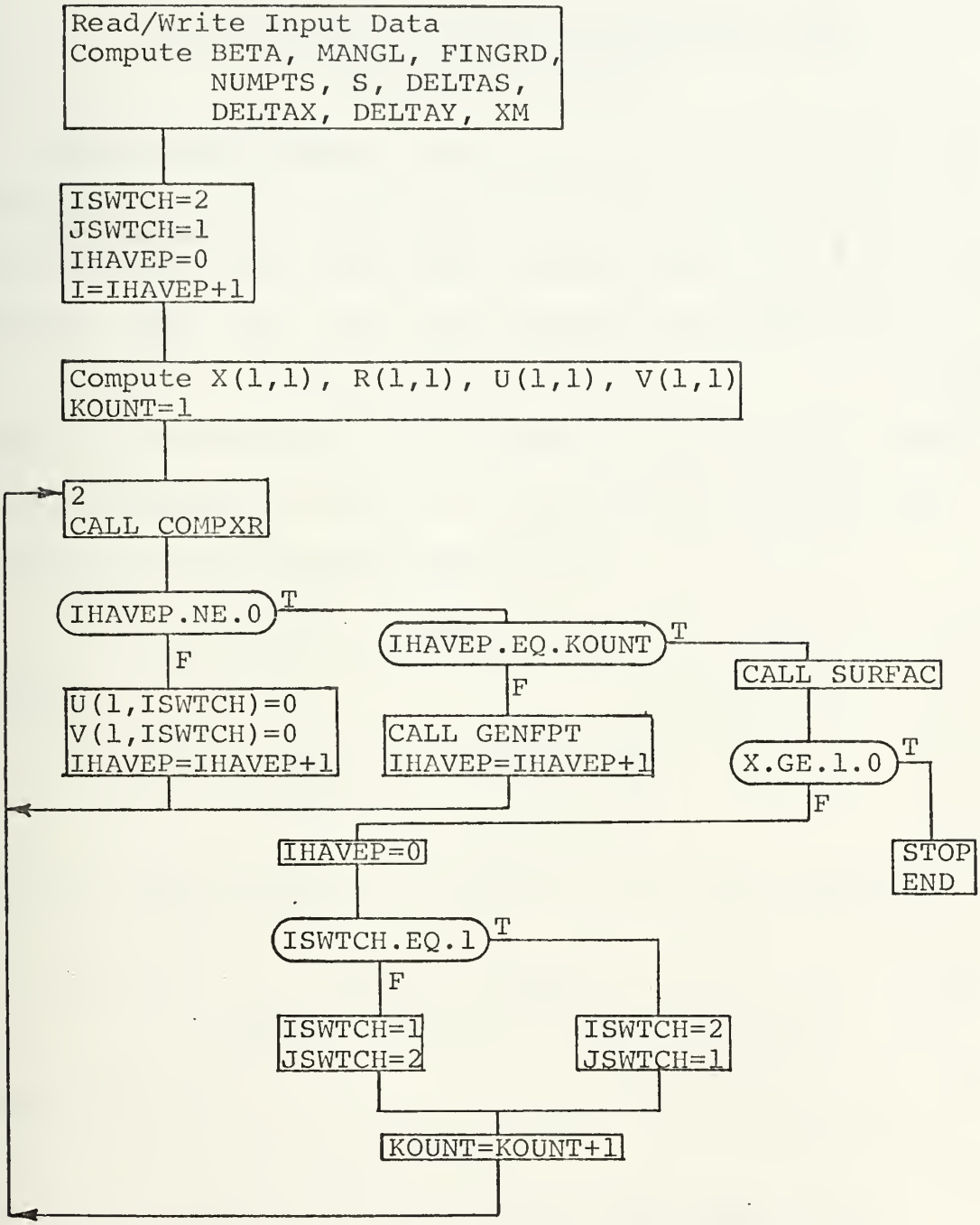
The following variables in the program take on the values defined below. All are floating point quantities unless otherwise stated. The dimensioned variables are listed first.

CP(200)	The pressure coefficient.
DELX,DELR(200)	The incremental distances between grid points on a right-running characteristic.
U(200,2)	The streamwise velocity perturbation.
V(200,2)	The product vr.
X,R(200,2)	The x and r coordinates of the grid point.
A,B	Used as factors in the equations for u and v.

BETA	The square root of the quantity M^2-1.
DELTAS	The incremental distance between adjacent grid points along the initial Mach line.
DELTAX	The change in x from one point to the next on the initial Mach line.
DELTAY	The change in r from one point to the next on the initial Mach line.
FINGRD	Floating point NGFN.
HO	The maximum amplitude of the deflection.
MACH	Free-stream Mach number.
MANGL	The Mach angle, α.
MFREQ	The axial mode number, an integer.
NGFN	The grid fineness ratio, an even integer, less than 200.
NUMPTS	The number of grid points on the initial Mach line.
PI	3.141593
RETA	r_η
RO	The radius to length ratio.
RZETA	r_ξ
S	The length of the initial Mach line.
SLP	Slope of the surface at a surface grid point.
XM	Floating point MFREQ.
XMAX	The value of x at a surface grid point.
YO	The intercept of a right-running Mach line with the r-axis.

APPENDIX F

SAUER-HEINZ FLOW DIAGRAM



APPENDIX G

SUPERSONIC FLOW PAST AN INFINITELY LONG WAVY WALL CYLINDER

Leonard and Hedgepeth (Ref. 4) gave a solution for supersonic flow past an infinitely long wavy wall cylinder. Detailed computations using this solution were carried out by Anderson (Ref. 14). For the cylinder wall deflection

$$h = h_0 \sin m\pi x \cos n\theta \tag{G-1}$$

where m is the number of half-waves per unit axial length and n indicates the number of circumferential waves, one obtains for the pressure distribution.

$$Cp = 2h_0 \frac{m\pi}{M} A_1 \sin(m\pi x + \Psi_1) \tag{G-2}$$

where

$$A_1 e^{i\Psi_1} = \frac{M}{\sqrt{M^2-1}} \frac{H_n^{\,2}[m\pi R(M^2-1)^{\frac{1}{2}}]}{H_n^{\,2\prime}[m\pi R(M^2-1)^{\frac{1}{2}}]} \tag{G-3}$$

For the actual computation equation (G-3) was rewritten as

$$A_1 e^{i\Psi_1} = \frac{J_n(\gamma) - i\,Y_n(\gamma)}{[J_{n-1}(\gamma)-J_{n+1}(\gamma)]-i[Y_{n-1}(\gamma)-Y_{n+1}(\gamma)]} \cdot \frac{2M}{\sqrt{M^2-1}} \tag{G-4}$$

where

$$\gamma = m\pi R \sqrt{M^2-1} \tag{G-5}$$

For the case n = 0 the equation was rewritten as

$$A_1 e^{i\Psi_1} = \frac{J_0(\gamma) - i\,Y_o(\gamma)}{-J_1(\gamma) + i\,Y_1(\gamma)} \cdot \frac{M}{\sqrt{M^2-1}} \tag{G-6}$$

and

$$A_1 = |A_1 e^{i\Psi_1}| \tag{G-7}$$

$$\Psi_1 = \tan^{-1}\left[\frac{\mathrm{Im}(A_1 e^{i\Psi_1})}{\mathrm{Re}(A_1 e^{i\Psi_1})}\right] \tag{G-8}$$

The program listing appears in Appendix I.

APPENDIX H

TWO BLADE SOLUTION USING THE METHOD OF SINGULARITIES

As shown by Garrick and Rubinow (Ref. 12), the velocity potential of a single blade oscillating in supersonic flow is

$$\phi(x,y)=\frac{-1}{\sqrt{M^2-1}}\int_0^{x-y\sqrt{M^2-1}} v(\xi)e^{-ik\frac{M^2}{M^2-1}(x-\xi)}J_0\left[\frac{kM}{M^2-1}\sqrt{(x-\xi)^2-(M^2-1)y^2}\right]d\xi \qquad \text{(H-1)}$$

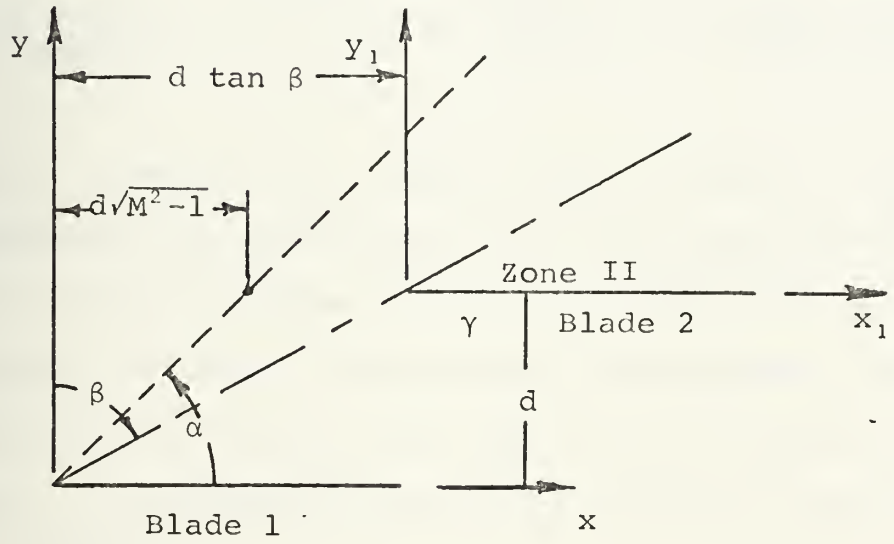


Figure H-1. Method of Singularities Configuration

The solution for supersonic flow past two oscillating blades in Zone II, (Figure H-1), can be obtained by superposition of single blade solutions. For this purpose the disturbed flow field in Zone II is written as the linear superposition of the single blade solutions for blades 1 and 2. However, in order to satisfy the flow tangency condition on the upper surface

of blade 2, a third single blade solution with unknown normal velocity must be added. Hence, one can write

$$\phi_T = \phi_1 + \phi_2 + \phi_{IP} \tag{H-2}$$

where

$$\phi_1 = \frac{-1}{\sqrt{M^2-1}} \int_0^{x-y\sqrt{M^2-1}} v(\xi) e^{-ik\frac{M^2}{M^2-1}(x-\xi)} J_0\left[\frac{kM}{M^2-1}\sqrt{(x-\xi)^2-(M^2-1)y^2}\right] d\xi \tag{H-3}$$

$$\phi_2 = \frac{-1}{\sqrt{M^2-1}} \int_0^{x_1-y_1\sqrt{M^2-1}} v(\xi) e^{-i\frac{kM^2}{M^2-1}(x_1-\xi)} J_0\left[\frac{kM}{M^2-1}\sqrt{(x_1-\xi)^2-(M^2-1)y_1^2}\right] d\xi \tag{H-4}$$

$$\phi_{IP} = \frac{-1}{\sqrt{M^2-1}} \int_0^{x_1-y_1\sqrt{M^2-1}} v_\sigma(\xi) e^{-i\frac{kM^2}{M^2-1}(x_1-\xi)} J_0\left[\frac{kM}{M^2-1}\sqrt{(x_1-\xi)^2-(M^2-1)y_1^2}\right] d\xi \tag{H-5}$$

where v_σ denotes the unknown normal velocity. Equation (H-3) represents the disturbance due to the first blade, equation (H-4) the disturbance due to the second blade alone, and equation (H-5) the interference disturbance. Note that the blade 2 and interference potentials are expressed in terms of an x_1, y_1 coordinate system with origin at the leading edge of blade 2.

Satisfying the boundary condition on the upper surface of blade 2 gives an equation for the unknown velocity, v_σ, i.e.,

$$v_\sigma(x,y) = \frac{1}{\sqrt{M^2-1}} \frac{\partial}{\partial y} \int_0^{x-y\sqrt{M^2-1}} v(\xi) e^{-i\frac{kM^2}{M^2-1}(x-\xi)} \cdot J_0\left[\frac{Mk}{M^2-1}\sqrt{(x-\xi)^2-(M^2-1)y^2}\right] d\xi \tag{H-6}$$

at $y = d$.

Now setting

$$v(\xi) = ik\, Z(\xi) + Z'(\xi) \tag{H-7}$$

and assuming the blades to pitch about their leading edges, i.e.,

$$Z(\xi) = \xi \tag{H-8}$$

and then performing the differentiation, one obtains

$$v_\sigma(x,y) = \frac{1}{\sqrt{M^2-1}} \left\{ \int_0^{x-y\sqrt{M^2-1}} [ikZ(\xi)+Z'(\xi)]\, e^{-i\frac{kM^2}{M^2-1}(x-\xi)} \right.$$

$$\cdot \frac{kMy}{\sqrt{(x-\xi)^2-(M^2-1)y^2}} \cdot J_1\left[\frac{kM}{M^2-1}\sqrt{(x-\xi)^2-(M^2-1)y^2}\right] d\xi$$

$$\left. - [ikZ(x-y\sqrt{M^2-1})+Z'(x-y\sqrt{M^2-1})] \cdot e^{-i\frac{kM^2}{M^2-1}y} \right\}$$

$$\text{at } y = d \tag{H-9}$$

Since, in general

$$C_P = ik\phi + \phi_x \tag{H-10}$$

the total pressure coefficient can be written as

$$C_P = C_{P1} + C_{P2} - C_{P_{IP}} \tag{H-11}$$

where C_{P1} is obtained from the single blade solution for blade 1 at y = d,

$$C_{P1} = \frac{k^2}{\sqrt{M^2-1}} \left(\int_0^{x-d\sqrt{M^2-1}} \left[-Z(\xi)+\frac{i}{k}Z'(\xi)\right] e^{-i\frac{kM^2}{M^2-1}(x-\xi)} \right.$$

$$\cdot J_0\left[\frac{kM}{M^2-1}\sqrt{(x-\xi)^2-(M^2-1)d^2}\right] d\xi - \frac{i}{k} \left\{ \int_0^{x-d\sqrt{M^2-1}} [-Z'(\xi)+\frac{i}{k}Z''(\xi)] \right.$$

$$\cdot e^{-i\frac{kM^2}{M^2-1}(x-\xi)} \cdot J_0\left[\frac{kM}{M^2-1}\sqrt{(x-\xi)^2-(M^2-1)d^2}\right] d\xi$$

$$\left. \left. + \frac{i}{k} e^{-i\frac{kM^2}{M^2-1}x} \cdot J_0\left[\frac{kM}{M^2-1}\sqrt{x^2-(M^2-1)d^2}\right] \right\} \right) \tag{H-12}$$

C_{P2} from the single blade solution for blade 2 at $y_1 = 0$,

$$C_{P2} = \frac{k^2}{\sqrt{M^2-1}}\left(\int_0^{x_1} [-Z(\xi)+\frac{i}{k}Z'(\xi)]e^{-i\frac{kM^2}{M^2-1}(x_1-\xi)}\right.$$

$$\cdot J_0[\frac{kM}{M^2-1}(x_1-\xi)]d\xi - \frac{i}{k}\left\{\int_0^{x_1}[-Z'(\xi)+\frac{i}{k}Z''(\xi)]e^{-i\frac{kM^2}{M^2-1}(x_1-\xi)}\right.$$

$$\left.\left.\cdot J_0[\frac{kM}{M^2-1}(x_1-\xi)]d\xi + \frac{i}{k}\, e^{-i\frac{kM^2}{M^2-1}x_1}\ J_0[\frac{kM}{M^2-1}x_1]\right\}\right) \qquad \text{(H-13)}$$

and

$$C_{P_{IP}} = ik\,\phi_{IP} + \phi_{IP} \qquad \text{(H-14)}$$

where

$$\phi_{IP} = \frac{-1}{\sqrt{M^2-1}}\int_0^{x_1} v_\sigma(\gamma)e^{-i\frac{kM^2}{M^2-1}(x_1-\gamma)}\ J_0[\frac{kM}{M^2-1}(x_1-\gamma)]d\gamma \qquad \text{(H-15)}$$

and

$$\phi_{IPx} = \frac{-1}{\sqrt{M^2-1}}\left\{\int_0^{x_1} v_\sigma(\gamma)[-i\frac{kM^2}{M^2-1}\ e^{-i\frac{kM^2}{M^2-1}(x_1-\xi)}\right.$$

$$\cdot J_0[\frac{kM}{M^2-1}(x_1-\gamma)]d\gamma - \int_0^{x_1} v_\sigma(\gamma)\ e^{-i\frac{kM}{M^2-1}(x_1-\gamma)}\cdot\frac{kM}{M^2-1}$$

$$\left.\cdot J_1[\frac{kM}{M^2-1}(x_1-\gamma)]d\gamma\right\} \qquad \text{(H-16)}$$

where

$$v_\sigma(x) = \frac{1}{\sqrt{M^2-1}}\left\{\int_0^{x-d\sqrt{M^2-1}}[ikZ(\xi)+Z'(\xi)]e^{-i\frac{kM^2}{M^2-1}(x-\xi)}\right.$$

$$\cdot \frac{kMd}{\sqrt{(x-\xi)^2-(M^2-1)d^2}}\cdot J_1[\frac{kM}{M^2-1}\sqrt{(x-\xi)^2-(M^2-1)d^2}]d\xi$$

$$\left.-[ikZ(x-d\sqrt{M^2-1}) + Z'(x-d\sqrt{M^2-1})]e^{-i\frac{kM^2}{M^2-1}d}\right\} \qquad \text{(H-17)}$$

Note that the substitution of $v_\sigma(x)$ into equations (H-15) and (H-16) for ϕ_{IP} and ϕ_{IPx} requires one to relate x and γ by

$$x = \gamma + d \tan \beta \qquad \text{(H-18)}$$

(see Figure H-1)

These integrals were computed using the Trapexoidal Rule and IBM subroutine BESJ. A listing of the program follows in Appendix I.

```
C                                                                       00000010
C              LINEARIZED METHOD OF CHARACTERISTICS                     00000020
C      FOR   SUPERSONIC FLOW PAST A VIBRATING CYLINDRICAL SHELL         00000030
C                                                                       00000040
       DIMENSION DATE(3)                                                00000050
       COMMON DPHII,DPHIK,PHI(400,2),PHK(400,2),QPHI(400,2),IPHI1,IPHI2,00000060
      4IKM,IKW,PARW,W,II,REALCP(400),CP(400),AMAGCP(400),               00000070
      1    MACH,K,MANGL,RADIUS,DELTAS,XLENGT,BETA,DSTSTR,HDSTRL,TRNGLH, 00000080
      2  X(400,2),R(400,2),NFREQ,FINGRD,ISWTCH,JSWTCH,ILINE,MLINE,JLINE,00000090
      3  MFREQ,IHAVEP,KOUNT                                             00000100
       INTEGER FINGRD                                                   00000110
       REAL    MACH,MANGL,K                                            00000120
       COMPLEX DPHII,DPHIK,PHI,IKM,CP,PX ,IPHI1,IPHI2,IKW,PHK,II,QPHI,PR 00000130
C                                                                       00000140
C      PRINT NAME OF PROGRAM AND DATE OF RUN                            00000150
C                                                                       00000160
       READ (5,10)DATE                                                  00000170
 10    FORMAT (3A4)                                                     00000180
 11    FORMAT ('1',/////,40X,'LINEARIZED METHOD OF CHARACTERISTICS'//40X,00000190
      1'FOR SUPERSONIC FLOW PAST A VIBRATING CYLINDRICAL SHELL'//20X     00000200
      2'DATE OF RUN-',3A4///)                                           00000210
       WRITE (6,11) DATE                                                00000220
C                                                                       00000230
C      READ INPUT                                                       00000240
C                                                                       00000250
       READ(5,1)MACH,RADIUS,K,MFREQ,NR,NFREQ,FINGRD                     00000260
     1 FORMAT(F20.8,2F10.3,4I5)                                         00000270
C                                                                       00000280
C      COMPUTE CONSTANTS                                                00000290
C                                                                       00000300
       II=(0.0,1.0)                                                     00000310
       BETA=SQRT(MACH**2-1.0)                                           00000320
       MANGL=ARSIN(1.0/MACH)                                            00000330
       XLENGT=1./(2.*COS(MANGL))                                        00000340
       HEIGHT=SIN(MANGL)*XLENGT                                         00000350
       DELTAS=XLENGT /FINGRD                                            00000360
       DSTSTR=DELTAS*COS(MANGL)*2.0                                     00000370
       HDSTRL=.5*DSTSTR                                                 00000380
       ML=IFIX(FINGRD+1.0)                                              00000390
       TRNGLH=HDSTRL*TAN(MANGL)                                         00000400
C                                                                       00000410
C      SET COUNTERS                                                     00000420
C                                                                       00000430
 20    ISWTCH=2                                                         00000440
       JSWTCH=1                                                         00000450
       IHAVEP=1                                                         00000460
       KCUNT=1                                                          00000470
C                                                                       00000480
```

```
C       COMPUTE INTIAL VALUES                                                   00000490
C                                                                               00000500
        I=IHAVEP                                                                00000510
        QPHI(I,JSWTCH)=(0.,0.)                                                  00000520
        PHK(I,JSWTCH)=(0.,0.)                                                   00000530
        X(IHAVEP,JSWTCH)=0.0                                                    00000540
        R(IHAVEP,JSWTCH)=RADIUS                                                 00000550
        PI=3.141593                                                             00000560
        PARW=MFREQ*PI*COS(MFREQ*PI*X(IHAVEP,JSWTCH))                            00000570
        QR=PARW                                                                 00000580
        PHI(I,JSWTCH)=-2.0*QR/MACH                                              00000590
        PX=(MACH/(2.0*BETA))*(PHK(I,JSWTCH)+PHI(I,JSWTCH))                      00000600
        CP(KOUNT)=-2.0*(II*K*QPHI(I,JSWTCH)+PX)                                 00000610
        REALCP(KOUNT)=REAL(CP(KOUNT))                                           00000620
        AMAGCP(KOUNT)=AIMAG(CP(KOUNT))                                          00000630
C                                                                               00000640
CC      WRITE OUT INPUT                                                         00000650
C                                                                               00000660
        WRITE (6,12) MACH,K,NFREQ,FINGRD ,RADIUS ,MFREQ                         00000670
 12     FORMAT (1H0,20X,'FREESTREAM MACH NUMBER =',F9.6//20X,'REDUCED FREQ00000680
     1='',F9.5//20X,'RAD MODE NO. N   =',I2//20X,'GRID FINENESS IS = ',  00000690
     2 I3  //20X,'CYLINDER RADIUS IS =',F9.2//20X,'AXIAL MODE NO. M   ='00000700
     3,I2)                                                                      00000710
        KCUNT=2                                                                 00000720
C                                                                               00000730
CC      COMPXR COMPUTES THE VALUE FOR X AND R GIVEN IHAVEP,ISWTCH               00000740
C                                                                               00000750
 104    CALL COMPXR                                                             00000760
        PARW=MFREQ*PI*COS(MFREQ*PI*X(IHAVEP,ISWTCH))                            00000770
        W=SIN(MFREQ*PI*X(IHAVEP,ISWTCH))                                        00000780
        XKM=(K*MACH)/BETA                                                       00000790
        IKM=II*XKM                                                              00000800
        XKW=K*W                                                                 00000810
        IKW=II*XKW                                                              00000820
C                                                                               00000830
CC      THE PROGRAM NOW GOES TO 30 IF POINT IS ON INITIAL MACH LINE.            00000840
C                                                                               00000850
 1001   IF (IHAVEP.EQ.1) GO TO 30                                               00000860
C                                                                               00000870
CC      TEST WHETHER THIS IS A SURFACE POINT.                                   00000880
C                                                                               00000890
        IF  (IHAVEP.EQ.KOUNT) GO TO 40                                          00000900
        GO TO 60                                                                00000910
C                                                                               00000920
 30     CALL MACHLN                                                             00000930
        GO TO 50                                                                00000940
C                                                                               00000950
 40     CALL PANPNT                                                             00000960
```

```
      GO TO 50                                                          00000970
C                                                                       00000980
 60   CALL GENFPT                                                       00000990
   50 A=IHAVEP                                                          00001000
      B=ISWTCH                                                          00001010
      PX=(MACH/(2.0*BETA))*(PHK(A,B)+ PHI(A,B))                         00001020
      PR=(MACH/2.0)*(PHK(A,B)-PHI(A,B))                                 00001030
 1000   IF (IHAVEP.EQ.KOUNT) GO TO 25                                   00001040
      GO TO 35                                                          00001050
C                                                                       00001060
C     INCREMENT FOR NEXT LINES                                          00001070
C                                                                       00001080
 25   IF (ISWTCH.EQ.1.) GO TO 103                                       00001090
      ISWTCH=1                                                          00001100
      JSWTCH=2                                                          00001110
      IHAVEP=1                                                          00001120
      GO TO 108                                                         00001130
 103  ISWTCH=2                                                          00001140
      JSWTCH=1                                                          00001150
      IHAVEP=1                                                          00001160
      GO TO 108                                                         00001170
C                                                                       00001180
C     AT 35 INCREMENT ALONG PRESENT LINE                                00001190
C                                                                       00001200
 35   IHAVEP=IHAVEP+1                                                   00001210
      GO TO 104                                                         00001220
 108   KOUNT=KOUNT+1                                                   00001230
      IF ((KOUNT-1).EQ.ML)GO TO 101                                     00001240
      GO TO 104                                                         00001250
  101 CONTINUE                                                          00001260
      CALL INTEG(REALCP,MFREQ,NR,DSTSTR,ML)                             00001270
      STOP                                                              00001280
      END                                                               00001290
C                                                                       00001300


      SUBROUTINE COMPXR                                                 00001310
C                                                                       00001320
      REAL   MACH,MANGL,K,KW                                            00001330
C                                                                       00001340
C     COMPUTES THE X AND R COORDINATES                                  00001350
C                                                                       00001360
      COMMON DPHII,DPHIK,PHI(400,2),PHK(400,2),QPHI(400,2),IPHI1,IPHI2, 00001370
     4IKM,IKW,PARW,W,II,REALCP(400),CP(400),AMAGCP(400),                00001380
     1    MACH,K,MANGL,RADIUS,DELTAS,XLENGT,BETA,DSTSTR,HDSTRL,TRNGLH,  00001390
     2  X(400,2),R(400,2),NFREQ,FINGRD,ISWTCH,JSWTCH,ILINE,MLINE,JLINE, 00001400
     3  MFREQ,IHAVEP,KOUNT                                              00001410
      COMPLEX DPHII,DPHIK,PHI,IKM,CP,PX ,IPHI1,IPHI2,IKW,PHK,II,QPHI,PR 00001420
```

```
      INTEGER FINGRD                                                    00001430
C                                                                       00001440
CC    TEST FOR POINT ON MACH LINE                                       00001450
C                                                                       00001460
      IF (IHAVEP.EQ.1) GO TO 10                                         00001470
      I=IHAVEP                                                          00001480
      J=IHAVEP-1                                                        00001490
      X(I,ISWTCH)=X(J      ,ISWTCH)+HDSTRL                              00001500
      R(I,ISWTCH)=R(J      ,ISWTCH)       -TRNGLH                       00001510
      RETURN                                                            00001520
   10 I=IHAVEP                                                          00001530
      X(I,ISWTCH)=X(I      ,JSWTCH)+HDSTRL                              00001540
      R(I,ISWTCH)=R(I      ,JSWTCH)+TRNGLH                              00001550
      RETURN                                                            00001560
      END                                                               00001570

      SUBROUTINE MACHLN                                                 00001580
      COMMON DPHII,DPHIK,PHI(400,2),PHK(400,2),QPHI(400,2),IPHI1,IPHI2, 00001590
     4IKM,IKW,PARW,W,II,REALCP(400),CP(400),AMAGCP(400),                00001600
     1    MACH,K,MANGL,RADIUS,DELTAS,XLENGT,BETA,DSTSTR,HDSTRL,TRNGLH,  00001610
     2  X(400,2),R(400,2),NFREQ,FINGRD,ISWTCH,JSWTCH,ILINE,MLINE,JLINE, 00001620
     3  MFREQ,IHAVEP,KOUNT                                              00001630
      REAL   MACH,MANGL,K,KW                                            00001640
      COMPLEX                AA2,                                       00001650
     1A,B,C,D,DPHII,DPHIK,PHI,IKM,CP,PX ,IPHI1,IPHI2,IKW,PHK,II,QPHI    00001660
     2 ,PR                                                              00001670
      INTEGER FINGRD                                                    00001680
      I=IHAVEP                                                          00001690
      PHK(I,ISWTCH)=(0.,0.)                                             00001700
      A=1.+DELTAS/(4.*R(I,ISWTCH)*MACH)+0.5*IKM*DELTAS                  00001710
      AA2=1.-DELTAS/ (4.*R(I,JSWTCH)*MACH)-0.5*IKM*DELTAS               00001720
      PHI(I,ISWTCH)=PHI(I,JSWTCH)*(AA2/A)                               00001730
      QPHI(I,ISWTCH)=(0.,0.)                                            00001740
   10 RETURN                                                            00001750
      END                                                               00001760

      SUBROUTINE GENFPT                                                 00001770
C                                                                       00001780
CC    GENFPT COMPUTES THE VELOCITY POTENTIAL AND ITS DERIVATIVES AT A   00001790
CC    GENERAL FLOW FIELD POINT.                                         00001800
C                                                                       00001810
      COMMON DPHII,DPHIK,PHI(400,2),PHK(400,2),QPHI(400,2),IPHI1,IPHI2, 00001820
     4IKM,IKW,PARW,W,II,REALCP(400),CP(400),AMAGCP(400),                00001830
     1    MACH,K,MANGL,RADIUS,DELTAS,XLENGT,BETA,DSTSTR,HDSTRL,TRNGLH,  00001840
     2  X(400,2),R(400,2),NFREQ,FINGRD,ISWTCH,JSWTCH,ILINE,MLINE,JLINE, 00001850
     3  MFREQ,IHAVEP,KOUNT                                              00001860
```

```
      REAL    MACH,MANGL,K,KW,N                                                 00001870
      COMPLEX                  AA1,AA2,AA3,                                     00001880
     1A,B,C,D,DPHII,DPHIK,PHI,IKM,CP,PX ,IPHI1,IPHI2,IKW,PHK,II,QPHI           00001890
     2  ,PR,E,F                                                                00001900
      INTEGER FINGRD                                                           00001910
C                                                                              00001920
      I=IHAVEP                                                                 00001930
      J=IHAVEP -1                                                              00001940
      N=FLOAT(NFREQ)                                                           00001950
C                                                                              00001960
      A=1.-DELTAS/(4.*R(I,ISWTCH)*MACH)+0.5*IKM*DELTAS                         00001970
C                                                                              00001980
      AA1=1.+DELTAS/(4.*R(J,ISWTCH)*MACH)-0.5*IKM*DELTAS                       00001990
C                                                                              00002000
      AA3=-1.+DELTAS/ (4.*R(I,JSWTCH)*MACH)+0.5*IKM*DELTAS                     00002010
C                                                                              00002020
      B=DELTAS/(4.*R(I,ISWTCH)*MACH)+0.5*IKM*DELTAS-0.25*DELTAS**2*            00002030
     1   (K**2-(N**2/(R(I,ISWTCH)**2*MACH**2)))                                00002040
C                                                                              00002050
C                                                                              00002060
      C=PHK(J,ISWTCH)*AA1+PHI(J,ISWTCH)*(-DELTAS/(4.*R(J,ISWTCH)*MACH)         00002070
     1 -0.5*IKM*DELTAS+0.25*DELTAS**2*(K**2-(N**2/(R(I,ISWTCH)**2*MACH**      00002080
     22))))+ QPHI(J,ISWTCH)*DELTAS*0.5*(K**2- N**2/(R(J,ISWTCH)**2*MACH       00002090
     3**2))+    QPHI(J,ISWTCH)*DELTAS*0.5*(K**2- N**2/(R(I,ISWTCH)**2*       00002100
     4 MACH**2))                                                              00002110
C                                                                              00002120
C                                                                              00002130
      D=DELTAS/(4.0  *R(I,ISWTCH)*MACH)-0.5  *IKM*DELTAS+0.25  *DELTAS**      00002140
     12*(K**2-(N**2/(R(I,ISWTCH)**2*MACH**2)))                                00002150
C                                                                              00002160
C                                                                              00002170
      F=PHI(I,JSWTCH)*AA3+PHK(I,JSWTCH)*(-DELTAS/(4.*R(I,JSWTCH)*MACH)         00002180
     1 +0.5*IKM*DELTAS-0.25*DELTAS**2*(K**2-(N**2/(R(I,ISWTCH)**2*MACH**      00002190
     22))))- QPHI(I,JSWTCH)*DELTAS*0.5*(K**2- N**2/(R(I,JSWTCH)**2*MACH       00002200
     3**2))-    QPHI(I,JSWTCH)*DELTAS*0.5*(K**2- N**2/(R(I,ISWTCH)**2*       00002210
     4 MACH**2))                                                              00002220
C                                                                              00002230
      E=-1.-DELTAS/(4.*R(I,ISWTCH)*MACH)-0.5*IKM*DELTAS                        00002240
C                                                                              00002250
C     CRAMER'S RULE                                                            00002260
C                                                                              00002270
      PHK(I,ISWTCH)=(C*E-B*F)/(A*E -B*D)                                       00002280
      PHI(I,ISWTCH)=(F*A-C*D)/(A*E -B*D)                                       00002290
C                                                                              00002300
C     TRAPAZOIDAL RULE TO INTEGRATE.                                           00002310
C                                                                              00002320
      IPHI1=QPHI(I,JSWTCH)+0.5*(PHK(I,JSWTCH)+PHK(I,ISWTCH))*DELTAS            00002330
      IPHI2=QPHI(J,ISWTCH)+0.5*(PHI(J,ISWTCH)+PHI(I,ISWTCH))*DELTAS            00002340
```

```
C                                                                       00002350
CC      AVERAGE THE TWO                                                 00002360
C                                                                       00002370
      QPHI(I,ISWTCH)=0.5*(IPHI1+IPHI2)                                  00002380
 15   RETURN                                                            00002390
      END                                                               00002400

      SUBROUTINE PANPNT                                                 00002410
C                                                                       00002420
CC      PANPNT COMPUTES THE VALUES OF PHI'S AND QHPI AT A SURFACE POINT 00002430
C                                                                       00002440
      COMMON DPHII,DPHIK,PHI(400,2),PHK(400,2),QPHI(400,2),IPHI1,IPHI2, 00002450
     4IKM,IKW,PARW,W,II,REALCP(400),CP(400),AMAGCP(400),                00002460
     1    MACH,K,MANGL,RADIUS,DELTAS,XLENGT,BETA,DSTSTR,HDSTRL,TRNGLH,  00002470
     2  X(400,2),R(400,2),NFREQ,FINGRD,ISWTCH,JSWTCH,ILINE,MLINE,JLINE, 00002480
     3  MFREQ,IHAVEP,KOUNT                                             00002490
      REAL    MACH,MANGL,K,KW,N                                         00002500
      COMPLEX                    AA1,AA2                                00002510
     1A,B,C,D,DPHII,DPHIK,PHI,IKM,CP,PX ,IPHI1,IPHI2,IKW,PHK,II,QPHI    00002520
     2 ,PR                                                              00002530
      INTEGER FINGRD                                                    00002540
      N=FLOAT(NFREQ)                                                    00002550
      I=IHAVEP                                                          00002560
      J=IHAVEP-1                                                        00002570
      A=1.-DELTAS/(4.*R(I,ISWTCH)*MACH)+0.5*IKM*DELTAS                  00002580
      AA1=1.+DELTAS/(4.*R(J,ISWTCH)*MACH)-0.5*IKM*DELTAS                00002590
      B=DELTAS/(4.*R(I,ISWTCH)*MACH)+0.5*IKM*DELTAS-0.25*DELTAS**2*     00002600
     1  (K**2-(N**2/(R(I,ISWTCH)**2*MACH**2)))                          00002610
      C=PHK(J,ISWTCH)*AA1+PHI(J,ISWTCH)*(-DELTAS/(4.*R(J,ISWTCH)*MACH)  00002620
     1 -0.5*IKM*DELTAS+0.25*DELTAS**2*(K**2-(N**2/(R(I,ISWTCH)**2*MACH**00002630
     22))))+ QPHI(J,ISWTCH)*DELTAS*0.5*(K**2- N**2/(R(J,ISWTCH)**2*MACH 00002640
     3**2))+   QPHI(J,ISWTCH)*DELTAS*0.5*(K**2- N**2/(R(I,ISWTCH)**2*  00002650
     4 MACH**2))                                                        00002660
C                                                                       00002670
C                                                                       00002680
      PHK(I,ISWTCH)=       (C+B*(2.0/MACH)*(PARW+IKW))/(A+B)            00002690
      PHI(I,ISWTCH)=       (-A*(2.0/MACH)*(PARW+IKW)+C)/(A+B)           00002700
C                                                                       00002710
      QPHI(I     ,ISWTCH)=QPHI(J,ISWTCH)+0.5*(PHI(J,ISWTCH)+PHI(I     ,  00002720
     1  ISWTCH))*DELTAS                                                 00002730
      PX=(MACH/(2.0*BETA))*(PHK(I,ISWTCH)+PHI(I,ISWTCH))                00002740
      CP(KOUNT)=-2.0*(II*K*QPHI(I,ISWTCH)+PX)                           00002750
      REALCP(KOUNT)=REAL(CP(KOUNT))                                     00002760
      AMAGCP(KOUNT)=AIMAG(CP(KOUNT))                                    00002770
 10   RETURN                                                            00002780
      END                                                               00002790
```

```
      SUBROUTINE INTEG(Y,M,NR,DX,ML)                                    00002800
      DIMENSION Y(ML)                                                   00002810
      Q=0.0                                                             00002820
      X1=0.0                                                            00002830
      RN=FLOAT(NR)                                                      00002840
      PSI1=0.0                                                          00002850
      PROD1=0.0                                                         00002860
      DO 1 I=2,ML                                                       00002870
      X2=X1+DX                                                          00002880
      PSI2=SIN(RN*3.141593*X2)                                          00002890
      PROD2=PSI2*Y(I)                                                   00002900
      Q=Q+(PROD2+PROD1)*DX/2.0                                          00002910
      PROD1=PROD2                                                       00002920
      X1=X2                                                             00002930
    1 CONTINUE                                                          00002940
      Q=Q/2.0                                                           00002950
      WRITE(6,100)M,NR,Q                                                00002960
  100 FORMAT(' ',10X,'Q(',I2,',',I2,')R=',E13.7,///)                    00002970
  101 FORMAT('0','GENERALIZED AERODYNAMIC FORCE:',/)                    00002980
      RETURN                                                            00002990
      END                                                               00003000
```

```
C                                                                       00000010
C                    ONE DEGREE OF FREEDOM CASCADE                      00000020
C                                                                       00000030
      DIMENSION DATE(3)                                                 00000040
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(400 00000050
     1,2),C22I(400,2),X(400,2),Y(400,2)                                 00000060
      DIMENSION U33R(400,2),U33I(400,2),V33R(400,2),V33I(400,2),C33R(400 00000070
     1,2),C33I(400,2)                                                   00000080
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUB0,TNWDST,FMANGL,XLNGTH,DELTAS 00000090
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,   00000100
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,    00000110
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                           00000120
      COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I                          00000130
      COMMON/PCOR/ICO                                                   00000140
      COMMON/STG/STGR                                                   00000150
      COMMON/BLADE/NUM,NUMBLD                                           00000160
      COMMON/BLK6/W                                                     00000170
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                00000180
      INTEGER OLDL                                                      00000190
C                                                                       00000200
CC    PRINT NAME OF PROGRAM/DATE OF RUN.                                00000210
C                                                                       00000220
      READ(5,100)DATE                                                   00000230
  100 FORMAT(3A4)                                                       00000240
      WRITE(6,101)DATE                                                  00000250
  101 FORMAT(1H1,14(/),14(/),37X,51H OSCILLATING CASCADE PROGRAM        00000260
     1   ---RUN OF---,3A4,/,1H1)                                        00000270
C                                                                       00000280
CC    PRINT OUT ALL INPUT INFORMATION VIA INPUT.                        00000290
C                                                                       00000300
      CALL INPUT                                                        00000310
      OLDL=0                                                            00000320
      ICC=0                                                             00000330
C                                                                       00000340
CC    INITIALIZE FLOW QUANTITIES AND LOGIC VARIABLES.                   00000350
C                                                                       00000360
      CALL INTIAL                                                       00000370
      FAC=1.0/DSTSTR                                                    00000380
      NEWDST=FAC                                                        00000390
      XX=FAC-NEWDST                                                     00000400
      IF(XX.GE.0.5)NEWDST=NEWDST+1                                      00000410
      LIMIT=(NUMBLD-1)*NSTPTS+NEWDST                                    00000420
      LIMW=NEWDST                                                       00000430
      MAXI=NGRDFN+NSTPTS                                                00000440
      NUM=0                                                             00000450
      NUMOLD=0                                                          00000460
      NUMPI=1                                                           00000470
 1000 CONTINUE                                                          00000480
```

```
      CALL COMPXY                                                       00000490
C                                                                       00000500
CC    IF POINT IS ON THE INITIAL RIGHT-RUNNING MACH LINE, CALL MACHLN.  00000510
C                                                                       00000520
      IF(LCOUNT.EQ.0)GO TO 1                                            00000530
C                                                                       00000540
CC    IF POINT IS ON A BLADE OR IN A WAKE, GO TC 2.                     00000550
C                                                                       00000560
      IF(IHAVEP.EQ.LCOUNT)GO TO 2                                       00000570
C                                                                       00000580
CC    IF POINT IS ON INITIAL LEFT-RUNNING MACH LINE, CALL MACHLN.       00000590
C                                                                       00000600
      IF(IHAVEP.EQ.0)GO TO 1                                            00000610
C                                                                       00000620
CC    CONDITICN FOR FIRST ENCOUNTER WITH A BLADE.                       00000630
C                                                                       00000640
      IF((IHAVEP.EQ.NUMPI*NSTPTS).AND.(JLINE.EQ.MAXI))GO TO 7           00000650
      CALL GENFPT                                                       00000660
      IF(IHAVEP.EQ.LIMIT)GO TO 5                                        00000670
      IHAVEP=IHAVEP+1                                                   00000680
      GO TO 1000                                                        00000690
    1 CALL MACHLN                                                       00000700
      IF(IHAVEP.EQ.LIMIT)GO TO 5                                        00000710
      IHAVEP =IHAVEP+1                                                  00000720
      GO TO 1000                                                        00000730
    2 ICC=1                                                             00000740
C                                                                       00000750
CC    IF POINT IS IN A WAKE, CALL WAKE.                                 00000760
C                                                                       00000770
      IF(IHAVEP.GE.LIMW)GC TO 4                                         00000780
      CALL TOP                                                          00000790
      CALL BOTTCM                                                       00000800
      IF(IHAVEP.EQ.LIMIT)GO TO 8                                        00000810
      IHAVEP=IHAVEP+1                                                   00000820
      IF(NUM.EQ.0)GO TO 1000                                            00000830
      NUM=NUM-1                                                         00000840
      LCOUNT=LCOUNT+NGRDFN                                              00000850
      LIMW=LIMW-NSTPTS                                                  00000860
      GO TO 1000                                                        00000870
    4 CALL WAKE                                                         00000880
      IF(IHAVEP.EQ.LIMIT)ICO=0                                          00000890
C                                                                       00000900
CC    IF THIS IS THE LAST POINT TO BE COMPUTED IN THE ENTIRE FLOW       00000910
CC    FIELD AREA, TERMINATE PROGRAM.                                    00000920
C                                                                       00000930
      IF((NUM.GE.NUMBLD-1).AND.(IHAVEP.EQ.LIMIT))GO TO 8                00000940
      IF(IHAVEP.EQ.LIMIT)GO TO 5                                        00000950
      IHAVEP=IHAVEP+1                                                   00000960
```

```
      IF(NUM.EQ.0)GO TO 1000                                                    00000970
      LCOUNT=LCOUNT+NGRDFN                                                      00000980
      NUM=NUM-1                                                                 00000990
      GO TO 1000                                                                00001000
    5 CONTINUE                                                                  00001010
C                                                                               00001020
CC    SWITCH LOGIC VARIABLES TO START THE COMPUTATION BACK AT THE               00001030
CC    INITIAL LEFT-RUNNING MACH LINE.                                           00001040
C                                                                               00001050
      IHAVEP=0                                                                  00001060
      NUM=NUMOLD                                                                00001070
      JLINE=JLINE+1                                                             00001080
      IF(ISWTCH.EQ.1)GO TO 9                                                    00001090
      ISWTCH=1                                                                  00001100
      JSWTCH=2                                                                  00001110
      GO TO 10                                                                  00001120
    9 ISWTCH=2                                                                  00001130
      JSWTCH=1                                                                  00001140
   10 CONTINUE                                                                  00001150
      IF(NUMOLD.GT.0)GO TO 6                                                    00001160
      LCOUNT=LCOUNT+1                                                           00001170
      GO TO 1000                                                                00001180
    6 LCOUNT=OLDL                                                               00001190
      LIMW=NUMOLD*NSTPTS+NEWDST                                                 00001200
      OLDL=OLDL+1                                                               00001210
      GO TO 1000                                                                00001220
    7 CALL NEWBLD                                                               00001230
      ICO=1                                                                     00001240
      NUMOLD=NUM                                                                00001250
      LCOUNT=IHAVEP+NGRDFN                                                      00001260
      JLINE=IHAVEP                                                              00001270
      IHAVEP=IHAVEP+1                                                           00001280
      OLDL=IHAVEP                                                               00001290
      NUMPI=NUM+1                                                               00001300
      MAXI=NGRDFN+NUMPI*NSTPTS                                                  00001310
      NUM=NUM-1                                                                 00001320
      GO TO 1000                                                                00001330
    8 CONTINUE                                                                  00001340
      STOP                                                                      00001350
      END                                                                       00001360


      SUBROUTINE INPUT                                                          00001370
C                                                                               00001380
CC    SUBROUTINE INPUT READS ALL INPUT.                                         00001390
C                                                                               00001400
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(400       00001410
     1,2),C22I(400,2),X(400,2),Y(400,2)                                         00001420
```

```
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS00001430
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,  00001440
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,   00001450
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                           00001460
      COMMON/STG/STGR                                                   00001470
      COMMON/BLADE/NUM,NUMBLD                                           00001480
      COMMON/BLK6/W                                                     00001490
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                00001500
      NAMELIST/NAM1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,STGANG,FAZE,NUMBLD00001510
      READ(5,NAM1)                                                      00001520
      WRITE(6,NAM1)                                                     00001530
      FMANGL=ARSIN(1.0/FSTRMN)                                          00001540
      XLNGTH=TNWDST/SIN(FMANGL)                                         00001550
      FNGDPT=FLOAT(NGRDFN)                                              00001560
      DELTAS=XLNGTH/FNGDPT                                              00001570
      TRNGLH=TNWDST/FNGDPT                                              00001580
      HDSTRL=DELTAS*COS(FMANGL)                                         00001590
      DSTSTR=HDSTRL*2.0                                                 00001600
      STAG=TNWDST*TAN(STGANG*0.1745329E-01)                             00001610
      STGR=STAG-(TNWDST/TAN(FMANGL))                                    00001620
      NSTPTS=STGR/DSTSTR                                                00001630
      IF(NSTPTS.LT.1)NSTPTS=1                                           00001640
      STGR=NSTPTS*DSTSTR                                                00001650
      STGANG=ATAN((STGR+(TNWDST/TAN(FMANGL)))/TNWDST)*57.29578          00001660
      STAG=STGR+(TNWDST/TAN(FMANGL))                                    00001670
      S=SQRT(FSTRMN*FSTRMN-1.0)                                         00001680
      W=FSTRMN*FSTRMN/(FSTRMN*FSTRMN-1.0)                               00001690
      DELTA=FAZE*0.1745329E-01                                          00001700
      WRITE(6,5)NGRDFN,FSTRMN,REDFRQ,TNWDST,XSUBO,STGANG,DSTSTR,FAZE,NUM00001710
     1BLD                                                               00001720
    5 FORMAT(////,28H GRID FINENESS INPUT NUMBER=,I10,///,24H FREESTREAM00001730
     1 MACH NUMBER=,F20.7,///,34H REDUCED(DIMENSIONLESS) FREQUENCY=,F20.00001740
     27,///,27H DISTANCE BETWEEN AIRFOILS=,F20.7,///,52H HORIZONTAL POSI00001750
     3TION(DIMENSIONLESS) OF ELASTIC AXIS=,F20.7,///,26H COMPATABLE STAG00001760
     4GER ANGLE=,F20.7,///,9H DELTA X=,F20.7,///25H UPPER AIRFOIL PHASE 00001770
     5LAG=,F20.7,///,1X,I2,' BLADES',/)                                 00001780
      WRITE(6,6)                                                        00001790
    6 FORMAT('1',10X,'VALUES OF TEIPEL COMPLEX PERTURBATION AMPLITUDES',00001800
     1///,1X,'BLD',T10,'X',T21,'UR',T42,'UI',T58,'VR',T73,'VI',T90,'CR',00001810
     2T107,'CI',/)                                                      00001820
      RETURN                                                            00001830
      END                                                               00001840
```

```
      SUBROUTINE INTIAL                                                 00001850
C                                                                       00001860
C     INTIAL INITIALIZES ALL FLOW FIELD QUANTITIES.                     00001870
C                                                                       00001880
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(400 00001890
     1,2),C22I(400,2),X(400,2),Y(400,2)                                 00001900
      DIMENSION U33R(400,2),U33I(400,2),V33R(400,2),V33I(400,2),C33R(400 00001910
     1,2),C33I(400,2)                                                   00001920
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUB0,TNWDST,FMANGL,XLNGTH,DELTAS00001930
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,   00001940
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,    00001950
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                           00001960
      COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I                          00001970
      COMMON/BLK6/W                                                     00001980
      COMMON/BLADE/NUM,NUMBLD                                           00001990
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                00002000
      AI=.5*REDFRQ*DSTSTR                                               00002010
      BI=AI*W/2.0                                                       00002020
C                                                                       00002030
C     SET UP INITIAL VALUES FOR U,V,AND C AT (0,0).                     00002040
C                                                                       00002050
   50 VRPANL=-1.0/S                                                     00002060
      VIPANL=REDFRQ*XSUB0/S                                             00002070
   60 U22R(1,1)=-VRPANL                                                 00002080
      U22I(1,1)=-VIPANL                                                 00002090
      V22R(1,1)=-U22R(1,1)                                              00002100
      V22I(1,1)=-U22I(1,1)                                              00002110
      C22R(1,1)=-U22R(1,1)                                              00002120
      C22I(1,1)=-U22I(1,1)                                              00002130
      X(1,1)=0.0                                                        00002140
      Y(1,1)=0.0                                                        00002150
      U33R(1,1)=VRPANL                                                  00002160
      U33I(1,1)=VIPANL                                                  00002170
      V33R(1,1)=VRPANL                                                  00002180
      V33I(1,1)=VIPANL                                                  00002190
      C33R(1,1)=-VRPANL                                                 00002200
      C33I(1,1)=-VIPANL                                                 00002210
      ISWTCH=1                                                          00002220
      JSWTCH=2                                                          00002230
      JLINE=0                                                           00002240
      IHAVEP=1                                                          00002250
      LCOUNT=0                                                          00002260
      WRITE(6,100)X(1,1),U22R(1,1),U22I(1,1),V22R(1,1),V22I(1,1),C22R(1,00002270
     11),C22I(1,1)                                                      00002280
      WRITE(6,100)X(1,1),U33R(1,1),U33I(1,1),V33R(1,1),V33I(1,1),C33R(1,00002290
     11),C33I(1,1)                                                      00002300
  100 FORMAT(2H I,7E16.7,/)                                             00002310
      RETURN                                                            00002320
```

```
      END                                                                   00002330

      SUBROUTINE MACHLN                                                     00002340
C                                                                           00002350
CC    MACHLN COMPUTES THE VALUES OF U,V,AND C ALONG THE INITIAL MACH        00002360
C     LINE AT THE GIVEN X.                                                  00002370
C                                                                           00002380
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(400    00002390
     1,2),C22I(400,2),X(400,2),Y(400,2)                                     00002400
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS    00002410
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,      00002420
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,       00002430
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                               00002440
      COMMON/BLK6/W                                                         00002450
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                    00002460
      I=IHAVEP+1                                                            00002470
      ARG=REDFRQ*W*X(I,ISWTCH)                                              00002480
      U=COS(ARG)                                                            00002490
      V=SIN(ARG)                                                            00002500
C                                                                           00002510
CC    IF POINT IS ON INITIAL RIGHT-RUNNING MACH LINE, GO TO 2.              00002520
C                                                                           00002530
      IF(IHAVEP.NE.0)GO TO 2                                                00002540
      U22R(1,ISWTCH)=-VRPANL*U-VIPANL*V                                     00002550
      U22I(1,ISWTCH)=-VIPANL*U+VRPANL*V                                     00002560
      V22R(1,ISWTCH)=-U22R(1,ISWTCH)                                        00002570
      V22I(1,ISWTCH)=-U22I(1,ISWTCH)                                        00002580
      C22R(1,ISWTCH)=-U22R(1,ISWTCH)                                        00002590
      C22I(1,ISWTCH)=-U22I(1,ISWTCH)                                        00002600
      RETURN                                                                00002610
    2 U22R(I,ISWTCH)=VRPANL*U+VIPANL*V                                      00002620
      U22I(I,ISWTCH)=VIPANL*U-VRPANL*V                                      00002630
      V22R(I,ISWTCH)=U22R(I,ISWTCH)                                         00002640
      V22I(I,ISWTCH)=U22I(I,ISWTCH)                                         00002650
      C22R(I,ISWTCH)=-U22R(I,ISWTCH)                                        00002660
      C22I(I,ISWTCH)=-U22I(I,ISWTCH)                                        00002670
      RETURN                                                                00002680
      END                                                                   00002690

      SUBROUTINE GENFPT                                                     00002700
C                                                                           00002710
CC    GENFPT COMPUTES U,V,AND C AT A GENERAL FIELD POINT.                   00002720
C                                                                           00002730
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(400    00002740
     1,2),C22I(400,2),X(400,2),Y(400,2)                                     00002750
      DIMENSION U33R(400,2),U33I(400,2),V33R(400,2),V33I(400,2),C33R(400    00002760
```

```
     1,2),C33I(400,2)                                                          00002770
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS       00002780
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,         00002790
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,          00002800
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                                  00002810
      COMMON/PCOR/ICO                                                          00002820
      COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I                                00002830
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                       00002840
      I=IHAVEP+1                                                              00002850
C                                                                             00002860
CC    'ICO' INSURES PROPER SELECTION OF FLOW QUANTITIES AT P21 IF GRID         00002870
CC    PCINT IS THE FIRST GRID POINT AFTER A BLADE GRID POINT.                  00002880
C                                                                             00002890
      IF(ICO.EQ.1)GO TO 1                                                     00002900
      GO TO 2                                                                 00002910
    1 A=U33R(IHAVEP,ISWTCH)                                                   00002920
      B=U33I(IHAVEP,ISWTCH)                                                   00002930
      C=C33R(IHAVEP,ISWTCH)                                                   00002940
      D=C33I(IHAVEP,ISWTCH)                                                   00002950
      ICO=0                                                                   00002960
      GO TO 10                                                                00002970
    2 A=U22R(IHAVEP,ISWTCH)                                                   00002980
      B=U22I(IHAVEP,ISWTCH)                                                   00002990
      C=C22R(IHAVEP,ISWTCH)                                                   00003000
      D=C22I(IHAVEP,ISWTCH)                                                   00003010
   10 K12R=U22R(IHAVEP,JSWTCH)+C22R(IHAVEP,JSWTCH)+AI*U22I(IHAVEP,JSWTCH00003020
     1)                                                                       00003030
      K12I=-AI*U22R(IHAVEP,JSWTCH)+U22I(IHAVEP,JSWTCH)+C22I(IHAVEP,JSWTC00003040
     1H)                                                                      00003050
      K34R=U22R(IHAVEP+1,JSWTCH)-V22R(IHAVEP+1,JSWTCH)+BI*(U22I(IHAVEP+100003060
     1,JSWTCH)-C22I(IHAVEP+1,JSWTCH))                                         00003070
      K34I=U22I(IHAVEP+1,JSWTCH)-V22I(IHAVEP+1,JSWTCH)-BI*(U22R(IHAVEP+100003080
     1,JSWTCH)-C22R(IHAVEP+1,JSWTCH))                                         00003090
      K56R=A+V22R(IHAVEP,ISWTCH)+BI*(B-D)                                     00003100
      K56I=B+V22I(IHAVEP,ISWTCH)-BI*(A-C)                                     00003110
      G1=.5*(K34R+K56R)                                                       00003120
      G2=BI*K12I                                                              00003130
      G3=1.0-AI*BI                                                            00003140
      G4=.5*(K34I+K56I)                                                       00003150
      G5=BI*K12R                                                              00003160
      G6=2.0*BI                                                               00003170
      G7=G3*G3+G6*G6                                                          00003180
      G8=G1-G2                                                                00003190
      G9=G4+G5                                                                00003200
      U22R(I,ISWTCH)=(G8*G3+G9*G6)/G7                                         00003210
      U22I(I,ISWTCH)=(-G8*G6+G9*G3)/G7                                        00003220
      V22R(I,ISWTCH)=.5*(K56R-K34R)                                           00003230
      V22I(I,ISWTCH)=.5*(K56I-K34I)                                           00003240
```

```
      C22R(I,ISWTCH)=K12R-U22R(I,ISWTCH)+U22I(I,ISWTCH)*AI                00003250
      C22I(I,ISWTCH)=K12I-U22I(I,ISWTCH)-U22R(I,ISWTCH)*AI                00003260
      RETURN                                                              00003270
      END                                                                 00003280

      SUBROUTINE TOP                                                      00003290
C                                                                         00003300
CC    TOP COMPUTES U,V, AND C AT AN UPPER SURFACE POINT.                  00003310
C                                                                         00003320
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(400  00003330
     1,2),C22I(400,2),X(400,2),Y(400,2)                                   00003340
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS  00003350
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,    00003360
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,     00003370
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                             00003380
      COMMON/BLADE/NUM,NUMBLD                                             00003390
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                  00003400
      XN=FLOAT(NUM)                                                       00003410
      J=IHAVEP                                                            00003420
      I=IHAVEP+1                                                          00003430
      XNEW=X(I,ISWTCH)-STAG*XN                                            00003440
C                                                                         00003450
CC    IF THIS IS THE FIRST BLADE OR IF THE INTERBLADE PHASE ANGLE IS      00003460
CCC   ZERO, SIMPLIFY THE NORMAL VELOCITY COMPUTATION.                     00003470
C                                                                         00003480
      IF((DELTA.LE.1.0E-05).OR.(NUM .EQ.0))GO TO 999                      00003490
      ARG=DELTA*XN                                                        00003500
      V22R(I,ISWTCH)=-(COS(ARG)-(XNEW-XSUBO)*REDFRQ*SIN(ARG))/S           00003510
      V22I(I,ISWTCH)=-(SIN(ARG)+(XNEW-XSUBO)*REDFRQ*COS(ARG))/S           00003520
      GO TO 90                                                            00003530
  999 V22R(I,ISWTCH)=-1.0/S                                               00003540
      V22I(I,ISWTCH)=(-REDFRQ*(XNEW-XSUBO))/S                             00003550
   90 K34R=V22R(I     ,ISWTCH)                                            00003560
      K34I=V22I(I     ,ISWTCH)                                            00003570
    1 K12R=U22R(J     ,JSWTCH)+C22R(J     ,JSWTCH)+AI*U22I(J     ,JSWTCH)  00003580
      K12I=U22I(J,JSWTCH)+C22I(J,JSWTCH)-AI*U22R(J,JSWTCH)                00003590
      K56R=U22R(J     ,ISWTCH)+V22R(J     ,ISWTCH)+BI*(U22I(J     ,ISWTCH)-00003600
     1C22I(J     ,ISWTCH))                                                00003610
      K56I=U22I(J     ,ISWTCH)+V22I(J     ,ISWTCH)-BI*(U22R(J     ,ISWTCH)-00003620
     1C22R(J     ,ISWTCH))                                                00003630
      G1=1.0-AI*BI                                                        00003640
      G2=2.0*BI                                                           00003650
      G3=G1*G1+G2*G2                                                      00003660
      G4=K56R-K34R-BI*K12I                                                00003670
      G5=K56I-K34I+BI*K12R                                                00003680
      U22R(I     ,ISWTCH)=(G4*G1+G5*G2)/G3                                00003690
      U22I(I     ,ISWTCH)=(-G4*G2+G5*G1)/G3                               00003700
```

```
      C22R(I     ,ISWTCH)=K12R-U22R(I     ,ISWTCH)+AI*U22I(I,ISWTCH)        00003710
      C22I(I     ,ISWTCH)=K12I-U22I(I     ,ISWTCH)-AI*U22R(I,ISWTCH)        00003720
  147 WRITE(6,145)NUM,XNEW,U22R(I,ISWTCH),U22I(I,ISWTCH),V22R(I,ISWTCH),00003730
     1V22I(I,ISWTCH),C22R(I,ISWTCH),C22I(I,ISWTCH)                          00003740
  145 FORMAT(2H T,I2,7E16.7,/)                                              00003750
      RETURN                                                                00003760
      END                                                                   00003770

      SUBROUTINE COMPXY                                                     00003780
C                                                                           00003790
CC    COMPXY COMPUTES THE GRID POINT'S X,Y POSITION.                        00003800
C                                                                           00003810
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(40000003820
     1,2),C22I(400,2),X(400,2),Y(400,2)                                     00003830
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS00003840
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,   00003850
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,    00003860
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                           00003870
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                    00003880
      I=IHAVEP+1                                                            00003890
C                                                                           00003900
CC    CHECK WHETHER GRID POINT IS ON THE INITIAL LEFT-RUNNING MACH LINE.00003910
C                                                                           00003920
      IF((IHAVEP.EQ.0).AND.(LCOUNT.NE.0))GO TO 1                            00003930
      X(I,ISWTCH)=X(IHAVEP,ISWTCH)+HDSTRL                                   00003940
      Y(I,ISWTCH)=Y(IHAVEP,ISWTCH)-TRNGLH                                   00003950
      RETURN                                                                00003960
C                                                                           00003970
C                                                                           00003980
    1 X(I,ISWTCH)=X(1,JSWTCH)+HDSTRL                                        00003990
      Y(I,ISWTCH)=Y(1,JSWTCH)+TRNGLH                                        00004000
      RETURN                                                                00004010
      END                                                                   00004020

      SUBROUTINE NEWBLD                                                     00004030
C                                                                           00004040
CC    NEWBLD COMPUTES U,V, AND C AT THE FIRST POSITION ON A BLADE           00004050
CC    ENCOUNTERED FOR THE FIRST TIME.                                       00004060
C                                                                           00004070
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(40000004080
     1,2),C22I(400,2),X(400,2),Y(400,2)                                     00004090
      DIMENSION U33R(400,2),U33I(400,2),V33R(400,2),V33I(400,2),C33R(40000004100
     1,2),C33I(400,2)                                                       00004110
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS00004120
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,   00004130
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,    00004140
```

```
      3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                                  00004150
       COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I                                 00004160
       COMMON/BLADE/NUM,NUMBLD                                                  00004170
       REAL K12R,K12I,K34R,K34I,K56R,K56I                                       00004180
       NUM=NUM+1                                                                00004190
       XN=FLOAT(NUM)                                                            00004200
       I=IHAVEP+1                                                               00004210
       K12R=U22R(IHAVEP,JSWTCH)+C22R(IHAVEP,JSWTCH)+AI*U22I(IHAVEP,JSWTCH00004220
      1)                                                                        00004230
       K12I=-AI*U22R(IHAVEP,JSWTCH)+U22I(IHAVEP,JSWTCH)+C22I(IHAVEP,JSWTC00004240
      1H)                                                                       00004250
   12  K56R=U22R(IHAVEP,ISWTCH)+V22R(IHAVEP,ISWTCH)+BI*(U22I(IHAVEP,ISWTC00004260
      1H)-C22I(IHAVEP,ISWTCH))                                                  00004270
       K56I=U22I(IHAVEP,ISWTCH)+V22I(IHAVEP,ISWTCH)-BI*(U22R(IHAVEP,ISWTC00004280
      1H)-C22R(IHAVEP,ISWTCH))                                                  00004290
C                                                                               00004300
CC     IF THE INTERBLADE PHASE ANGLE IS ZERO, SIMPLIFY THE NORMAL               00004310
CC     VELOCITY COMPUTATION.                                                    00004320
C                                                                               00004330
       IF(DELTA.LT.1.0E-05)GO TO 1                                              00004340
       V22R(I,ISWTCH)=-(COS(DELTA*XN)+XSUBO*REDFRQ*SIN(DELTA*XN))/S              00004350
       V22I(I,ISWTCH)=-(SIN(DELTA*XN)-XSUBO*REDFRQ*COS(DELTA*XN))/S              00004360
       GO TO 2                                                                  00004370
    1  V22R(I,ISWTCH)=-1.0/S                                                    00004380
       V22I(I,ISWTCH)=XSUBO*REDFRQ/S                                            00004390
    2  K34R=V22R(I,ISWTCH)                                                      00004400
       K34I=V22I(I,ISWTCH)                                                      00004410
       G1=1.0-AI*BI                                                             00004420
       G2=2.0*BI                                                                00004430
       G3=G1*G1+G2*G2                                                           00004440
       G4=K56R-K34R-BI*K12I                                                     00004450
       G5=K56I-K34I+BI*K12R                                                     00004460
       U22R(I,ISWTCH)=(G4*G1+G5*G2)/G3                                          00004470
       U22I(I,ISWTCH)=(-G4*G2+G5*G1)/G3                                         00004480
       C22R(I,ISWTCH)=K12R-U22R(I,ISWTCH)+AI*U22I(I,ISWTCH)                     00004490
       C22I(I,ISWTCH)=K12I-U22I(I,ISWTCH)-AI*U22R(I,ISWTCH)                     00004500
       K34R=U22R(I,JSWTCH)-V22R(I,JSWTCH)+BI*(U22I(I,JSWTCH)-C22I(I,JSWTC00004510
      1H))                                                                      00004520
       K34I=U22I(I,JSWTCH)-V22I(I,JSWTCH)-BI*(U22R(I,JSWTCH)-C22R(I,JSWTC00004530
      1H))                                                                      00004540
       V33R(I,ISWTCH)=V22R(I,ISWTCH)                                            00004550
       V33I(I,ISWTCH)=V22I(I,ISWTCH)                                            00004560
       K56R=V33R(I,ISWTCH)                                                      00004570
       K56I=V33I(I,ISWTCH)                                                      00004580
       G4=K56R+K34R-BI*K12I                                                     00004590
       G5=K56I+K34I+BI*K12R                                                     00004600
       U33R(I,ISWTCH)=(G4*G1+G5*G2)/G3                                          00004610
       U33I(I,ISWTCH)=(-G4*G2+G5*G1)/G3                                         00004620
```

```
      C33R(I,ISWTCH)=K12R-U33R(I,ISWTCH)+AI*U33I(I,ISWTCH)                    00004630
      C33I(I,ISWTCH)=K12I-U33I(I,ISWTCH)-AI*U33R(I,ISWTCH)                    00004640
      NUM=NUM+1                                                               00004650
      WRITE(6,100)NUM                                                         00004660
      NUM=NUM-1                                                               00004670
  100 FORMAT('0',10X,'BLADE #',I2,//)                                         00004680
  147 WRITE(6,145)X(I    ,ISWTCH),U22R(I    ,ISWTCH),U22I(I    ,ISWTCH),00004690
     1V22R(I    ,ISWTCH),V22I(I    ,ISWTCH),C22R(I    ,ISWTCH),C22I(I    00004700
     2 ,ISWTCH)                                                               00004710
      WRITE(6,145)X(I    ,ISWTCH),U33R(I    ,ISWTCH),U33I(I    ,ISWTCH),00004720
     1V33R(I    ,ISWTCH),V33I(I    ,ISWTCH),C33R(I    ,ISWTCH),C33I(I    00004730
     2 ,ISWTCH)                                                               00004740
  145 FORMAT(2H N,7E16.7,/)                                                   00004750
      RETURN                                                                  00004760
      END                                                                     00004770


      SUBROUTINE WAKE                                                         00004780
C                                                                             00004790
CC    WAKE COMPUTES U,V, AND C ABOVE AND BELOW THE SLIP PLANE AFT OF A        00004800
CC    BLADE.                                                                  00004810
C                                                                             00004820
      DIMENSION A(8,8),B(8)                                                   00004830
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(40000004840
     1,2),C22I(400,2),X(400,2),Y(400,2)                                       00004850
      DIMENSION U33R(400,2),U33I(400,2),V33R(400,2),V33I(400,2),C33R(40000004860
     1,2),C33I(400,2)                                                         00004870
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS00004880
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,  00004890
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,   00004900
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                                 00004910
      COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I                                00004920
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                       00004930
      I=IHAVEP                                                                00004940
      II=IHAVEP+1                                                             00004950
      J=ISWTCH                                                                00004960
      JJ=JSWTCH                                                               00004970
C                                                                             00004980
CC    COMPUTE RIGHT HAND SIDE OF EQUATIONS.                                   00004990
C                                                                             00005000
      B(1)=U22R(I ,JJ)+C22R(I ,JJ)+AI*U22I(I ,JJ)                             00005010
      B(2)=U22I(I ,JJ)+C22I(I ,JJ)-AI*U22R(I ,JJ)                             00005020
      B(3)=U22R(I ,J )+V22R(I ,J )+BI*(U22I(I ,J )-C22I(I ,J ))               00005030
      B(4)=U22I(I ,J )+V22I(I ,J )-BI*(U22R(I ,J )-C22R(I ,J ))               00005040
      B(5)=U33R(I ,JJ)+C33R(I ,JJ)+AI*U33I(I ,JJ)                             00005050
      B(6)=U33I(I ,JJ)+C33I(I ,JJ)-AI*U33R(I ,JJ)                             00005060
      B(7)=U22R(II,JJ)-V22R(II,JJ)+BI*(U22I(II,JJ)-C22I(II,JJ))               00005070
      B(8)=U22I(II,JJ)-V22I(II,JJ)-BI*(U22R(II,JJ)-C22R(II,JJ))               00005080
```

```
C                                                                       00005090
CC      SET UP MATRIX OF COEFFICIENTS.                                  00005100
C                                                                       00005110
        DO 1 K=1,8                                                      00005120
        DO 1 L=1,8                                                      00005130
        A(K,L)=0.0                                                      00005140
    1 CONTINUE                                                          00005150
        A(1,1)=1.0                                                      00005160
        A(1,7)=1.0                                                      00005170
        A(2,2)=1.0                                                      00005180
        A(2,8)=1.0                                                      00005190
        A(3,1)=1.0                                                      00005200
        A(3,5)=1.0                                                      00005210
        A(4,2)=1.0                                                      00005220
        A(4,6)=1.0                                                      00005230
        A(5,3)=1.0                                                      00005240
        A(5,7)=1.0                                                      00005250
        A(6,4)=1.0                                                      00005260
        A(6,8)=1.0                                                      00005270
        A(7,3)=1.0                                                      00005280
        A(8,4)=1.0                                                      00005290
        A(1,2)=-AI                                                      00005300
        A(5,4)=-AI                                                      00005310
        A(6,3)=AI                                                       00005320
        A(2,1)=AI                                                       00005330
        A(3,2)=-BI                                                      00005340
        A(4,7)=-BI                                                      00005350
        A(7,4)=-BI                                                      00005360
        A(8,7)=-BI                                                      00005370
        A(3,8)=BI                                                       00005380
        A(4,1)=BI                                                       00005390
        A(7,8)=BI                                                       00005400
        A(8,3)=BI                                                       00005410
        A(7,5)=-1.0                                                     00005420
        A(8,6)=-1.0                                                     00005430
        CALL SIMQ(A,B,8,KS)                                             00005440
        U22R(II,J  )=B(1)                                               00005450
        U22I(II,J  )=B(2)                                               00005460
        U33R(II,J  )=B(3)                                               00005470
        U33I(II,J  )=B(4)                                               00005480
        V22R(II,J  )=B(5)                                               00005490
        V22I(II,J  )=B(6)                                               00005500
        C22R(II,J  )=B(7)                                               00005510
        C22I(II,J  )=B(8)                                               00005520
        V33R(II,J  )=B(5)                                               00005530
        V33I(II,J  )=B(6)                                               00005540
        C33R(II,J  )=B(7)                                               00005550
        C33I(II,J  )=B(8)                                               00005560
```

```
      RETURN                                                                00005570
      END                                                                   00005580

      SUBROUTINE BOTTOM                                                     00005590
C                                                                           00005600
CC    BOTTOM COMPUTES U,V, AND C AT A LOWER SURFACE POINT.                  00005610
C                                                                           00005620
      DIMENSION U22R(400,2),U22I(400,2),V22R(400,2),V22I(400,2),C22R(400    00005630
     1,2),C22I(400,2),X(400,2),Y(400,2)                                     00005640
      DIMENSION U33R(400,2),U33I(400,2),V33R(400,2),V33I(400,2),C33R(400    00005650
     1,2),C33I(400,2)                                                       00005660
      COMMON/BLK1/NGRDFN,FSTRMN,REDFRQ,XSUBO,TNWDST,FMANGL,XLNGTH,DELTAS    00005670
     1,DSTSTR,HDSTRL,TRNGLH,U22R,U22I,V22R,V22I,C22R,C22I,X,Y,S,DELTA,      00005680
     2ISWTCH,JSWTCH,JLINE,IHAVEP,AI,BI,K12R,K12I,K34R,K34I,K56R,K56I,       00005690
     3VRPANL,VIPANL,STGANG,NSTPTS,LCOUNT,STAG                               00005700
      COMMON/BLK5/U33R,U33I,V33R,V33I,C33R,C33I                             00005710
      REAL K12R,K12I,K34R,K34I,K56R,K56I                                    00005720
      I=IHAVEP                                                              00005730
      J=JSWTCH                                                              00005740
      II=IHAVEP+1                                                           00005750
      JJ=ISWTCH                                                             00005760
      K12R=U33R(I ,J )+C33R(I ,J )+AI*U33I(I ,J )                           00005770
      K12I=U33I(I ,J )+C33I(I ,J )-AI*U33R(I ,J )                           00005780
      K34R=U22R(II,J )-V22R(II,J )+BI*(U22I(II,J )-C22I(II,J ))             00005790
      K34I=U22I(II,J )-V22I(II,J )-BI*(U22R(II,J )-C22R(II,J ))             00005800
      V33R(II,JJ)=V22R(II,JJ)                                               00005810
      V33I(II,JJ)=V22I(II,JJ)                                               00005820
      K56R=V33R(II,JJ)                                                      00005830
      K56I=V33I(II,JJ)                                                      00005840
      G1=1.0-AI*BI                                                          00005850
      G2=2.0*BI                                                             00005860
      G3=G1*G1+G2*G2                                                        00005870
      G4=K56R+K34R-BI*K12I                                                  00005880
      G5=K56I+K34I+BI*K12R                                                  00005890
      U33R( II , JJ    )=(G4*G1+G5*G2)/G3                                   00005900
      U33I(  II , JJ   )=(-G4*G2+G5*G1)/G3                                  00005910
      C33R(II,JJ)=K12R-U33R(II,JJ)+AI*U33I(II,JJ)                           00005920
      C33I(II,JJ)=K12I-U33I(II,JJ)-AI*U33R(II,JJ)                           00005930
      WRITE(6,100)X(II,ISWTCH),U33R(II,ISWTCH),U33I(II,ISWTCH),V33R(II,I    00005940
     1SWTCH),V33I(II,ISWTCH),C33R(II,ISWTCH),C33I(II,ISWTCH)                00005950
  100 FORMAT(2H B,7E16.7,/)                                                 00005960
      RETURN                                                                00005970
```

```
C                          THE SAUER-HEINZ PROCEDURE                          00000020
C                                                                             00000030
      REAL MACH,MANGL                                                         00000040
      DIMENSION X(200,2),R(200,2),U(200,2),V(200,2),CP(200)                   00000050
      COMMON/BLK1/X,R,U,V,ISWTCH,JSWTCH,IHAVEP,CP                             00000060
      COMMON/BLK3/BETA,PI,HO,XM,RO                                            00000070
      COMMON/BLK4/KOUNT,DELTAX,DELTAY                                         00000080
      COMMON/BLK5/MANGL                                                       00000090
      READ(5,100)MACH,HO,RO,MFREQ,NGFN                                        00000100
      WRITE(6,101)NGFN,MACH,MFREQ,HO,RO                                       00000110
C                                                                             00000120
CC    SET UP GRID AND LOGIC                                                   00000130
C                                                                             00000140
      PI=3.141593                                                             00000150
      BETA=SQRT(MACH**2-1.0)                                                  00000160
      MANGL=ARSIN(1.0/MACH)                                                   00000170
      FINGRD=FLOAT(NGFN)                                                      00000180
      NUMPTS=NGFN+1                                                           00000190
      S=0.5/COS(MANGL)                                                        00000200
      DELTAS=S/FINGRD                                                         00000210
      DELTAX=DELTAS*COS(MANGL)                                                00000220
      DELTAY=DELTAS*SIN(MANGL)                                                00000230
      XM=FLOAT(MFREQ)                                                         00000240
      ISWTCH=2                                                                00000250
      JSWTCH=1                                                                00000260
      IHAVEP=0                                                                00000270
      I=IHAVEP+1                                                              00000280
C                                                                             00000290
CC    INITIALIZE FLOW QUANTITIES                                              00000300
C                                                                             00000310
      X(I,JSWTCH)=0.0                                                         00000320
      R(I,JSWTCH)=RO                                                          00000330
      U(I,JSWTCH)=0.0                                                         00000340
      V(I,JSWTCH)=0.0                                                         00000350
      KCUNT=1                                                                 00000360
    2 CALL COMPXR                                                             00000370
C                                                                             00000380
CC    IF POINT IS NOT ON THE INITIAL MACH LINE, GO TO 3.                      00000390
C                                                                             00000400
    9 IF(IHAVEP.NE.0)GO TO 3                                                  00000410
      U(1,ISWTCH)=0.0                                                         00000420
      V(1,ISWTCH)=0.0                                                         00000430
      IHAVEP=IHAVEP+1                                                         00000440
      GO TO 2                                                                 00000450
C                                                                             00000460
CC    IF POINT IS ON THE SURFACE, CALL SURFAC.                                00000470
C                                                                             00000480
    3 IF(IHAVEP.EQ.KOUNT)GO TO 4                                              00000490
```

```
      CALL GENFPT                                                       00000500
      IHAVEP=IHAVEP+1                                                   00000510
      GO TO 2                                                           00000520
    4 CALL SURFAC                                                       00000530
C                                                                       00000540
CC    IF THIS IS THE LAST SURFACE POINT, TERMINATE THE PROGRAM.         00000550
C                                                                       00000560
      IF(X(IHAVEP+1,ISWTCH).GE.1.0)GO TO 7                              00000570
C                                                                       00000580
CC    AFTER MEETING SURFACE, SWITCH LOGIC VARIABLES TO GO BACK          00000590
CC    TO THE INITIAL MACH LINE.                                         00000600
C                                                                       00000610
      IHAVEP=0                                                          00000620
      IF(ISWTCH.EQ.1)GO TO 5                                            00000630
      ISWTCH=1                                                          00000640
      JSWTCH=2                                                          00000650
      GO TO 6                                                           00000660
    5 ISWTCH=2                                                          00000670
      JSWTCH=1                                                          00000680
    6 KOUNT=KOUNT+1                                                     00000690
      GO TO 2                                                           00000700
    7 CONTINUE                                                          00000710
  100 FORMAT(3F10.5,2I10)                                               00000720
  101 FORMAT('1',10X,'SUPERSONIC FLOW OVER A CYLINDRICAL SHELL: THE SAUE00000730
     1R-HEINZ PROCEDURE          ',//,15X,'GRID FINENESS=',I5,/,15X,'MACH00000740
     2 NUMBER=',F11.6,/,15X,'AXIAL MODE # =',I3   ,/,15X,'AMPLITUDE=',F1000000750
     3.5,/,15X,'RADIUS TO LENGTH RATIO=',F10.5,/)                       00000760
  102 FORMAT(F10.5)                                                     00000770
  103 FORMAT('0',                              ///,T6,'IHAVEP',T20,'X',T3000000780
     1,'R',T45,'U',T60,'V',T75,'CP',//)                                 00000790
      STOP                                                              00000800
      END                                                               00000810


      SUBROUTINE GENFPT                                                 00000820
C                                                                       00000830
CC    GENFPT COMPUTES THE FLOW QUANTITIES IN THE GENERAL FLOW FIELD.    00000840
C                                                                       00000850
      DIMENSION X(200,2),R(200,2),U(200,2),V(200,2),CP(200)             00000860
      COMMON/BLK1/X,R,U,V,ISWTCH,JSWTCH,IHAVEP,CP                       00000870
      COMMON/BLK3/BETA,PI,HO,XM,RC                                      00000880
      COMMON/BLK4/KOUNT,DELTAX,DELTAY                                   00000890
      I=IHAVEP+1                                                        00000900
      RETA=(R(IHAVEP,ISWTCH)+R(I,ISWTCH))/2.0                           00000910
      RZETA=(R(I,JSWTCH)+R(I,ISWTCH))/2.0                               00000920
      A=V(IHAVEP,ISWTCH)-V(I,JSWTCH)+BETA*(RETA*U(IHAVEP,ISWTCH)+RZETA*U00000930
     1(I,JSWTCH))                                                       00000940
      B=BETA*(RZETA+RETA)                                               00000950
```

```
      U(I,ISWTCH)=A/B                                                          00000960
      A=RZETA*V(IHAVEP,ISWTCH)+RETA*V(I,JSWTCH)-BETA*RZETA*RETA*(U(I,JSW00000970
     1TCH)-U(IHAVEP,ISWTCH))                                                   00000980
      B=RZETA+RETA                                                             00000990
      V(I,ISWTCH)=A/B                                                          00001000
      RETURN                                                                   00001010
      END                                                                      00001020

      SUBROUTINE SURFAC                                                        00001030
C                                                                              00001040
CC    SURFAC COMPUTES THE FLOW QUANTITIES AT THE SURFACE.                      00001050
CC                                                                             00001060
      DIMENSION X(200,2),R(200,2),U(200,2),V(200,2),CP(200)                    00001070
      COMMON/BLK1/X,R,U,V,ISWTCH,JSWTCH,IHAVEP,CP                              00001080
      COMMON/BLK3/BETA,PI,HO,XM,RO                                             00001090
      I=IHAVEP+1                                                               00001100
      SLP=HO*PI*XM*COS(XM*PI*X(I,ISWTCH))                                      00001110
C     IF(IHAVEP.EQ.1)GO TO 1                                                   00001120
      RETA=(R(IHAVEP,ISWTCH)+R(I,ISWTCH))/2.0                                  00001130
      A=BETA*RETA*U(IHAVEP,ISWTCH)+V(IHAVEP,ISWTCH)-R(I,ISWTCH)*SLP            00001140
      B=BETA*RETA+SLP*R(I,ISWTCH)                                              00001150
      U(I,ISWTCH)=A/B                                                          00001160
      GO TO 2                                                                  00001170
C                                                                              00001180
CC ********************************************************************      00001190
CC *     IF VALUES ARE NOT BEING COMPUTED AT THE SURFACE BUT AT A MEAN *      00001200
CC *     SURFACE POSITION, THE FOLLOWING CARDS DESCRIBE THE VALUE OF   *      00001210
C  *     RETA AND U.                                                   *      00001220
    1 H=R(2,ISWTCH)-RO                                                         00001230
      RETA=RO+(H/4.0)*(3.0+(1.0/(BETA*SLP)))                                   00001240
      A=-R(2,ISWTCH)*SLP                                                       00001250
      B=BETA*RETA+SLP*R(2,ISWTCH)                                              00001260
      U(2,ISWTCH)=A/B                                                          00001270
C *     TO USE THIS PROCEDURE, DELETE THE "C" IN COLUMN 1 OF THE CARD *      00001280
C *     WHICH FOLLOWS THE DESCRIPTION OF "SLP" ABOVE.                 *      00001290
CC ********************************************************************      00001300
CC                                                                             00001310
    2 V(I,ISWTCH)=SLP*(1.0+U(I,ISWTCH))*R(I,ISWTCH)                            00001320
      CP(IHAVEP)=-2.0*U(I,ISWTCH)-(V(I,ISWTCH)/R(I,ISWTCH))**2+(U(I,ISWT00001330
     1CH)*BETA)**2                                                             00001340
      WRITE(6,100)IHAVEP,X(I,ISWTCH),R(I,ISWTCH),U(I,ISWTCH),V(I,ISWTCH)00001350
     1,CP(IHAVEP)                                                              00001360
  100 FORMAT('0',T8,I3,T17,F7.5,T27,F9.6,T41,E11.5,T57,E11.5,T73,E11.5//00001370
     1)                                                                        00001380
      RETURN                                                                   00001390
      END                                                                      00001400
```

```
      SUBROUTINE COMPXR                                                 00001410
C                                                                       00001420
CC    COMPXR COMPUTES THE X,R POSITION OF THE GRID POINT AND DETERMINES 00001430
C     THE VALUE OF X AND R AT THE INTERSECTION OF THE RIGHT- RUNNING    00001440
C     MACH LINE WITH THE SURFACE.                                       00001450
C                                                                       00001460
      REAL MANGL                                                        00001470
      DIMENSION X(200,2),R(200,2),U(200,2),V(200,2),CP(200),DELX(200),DE00001480
     1LR(200)                                                           00001490
      COMMON/BLK1/X,R,U,V,ISWTCH,JSWTCH,IHAVEP,CP                       00001500
      COMMON/BLK3/BETA,PI,HO,XM,RO                                      00001510
      COMMON/BLK4/KOUNT,DELTAX,DELTAY                                   00001520
      COMMON/BLK5/MANGL                                                 00001530
C                                                                       00001540
CC    CHECK WHETHER POINT IS ON THE INITIAL MACH LINE.                  00001550
C                                                                       00001560
      IF(IHAVEP.EQ.0)GO TO 1                                            00001570
      I=IHAVEP+1                                                        00001580
C                                                                       00001590
CC    CHECK WHETHER THIS IS A SURFACE POINT.                            00001600
C                                                                       00001610
      IF(IHAVEP.EQ.KOUNT)GO TO 2                                        00001620
      X(I,ISWTCH)=X(IHAVEP,ISWTCH)+DELX(IHAVEP)                         00001630
      R(I,ISWTCH)=R(IHAVEP,ISWTCH)-DELR(IHAVEP)                         00001640
      RETURN                                                            00001650
    1 X(1,ISWTCH)=X(1,JSWTCH)+DELTAX                                    00001660
      R(1,ISWTCH)=R(1,JSWTCH)+DELTAY                                    00001670
      RETURN                                                            00001680
C                                                                       00001690
CC    COMPUTE THE POINT OF INTERSECTION.                                00001700
C                                                                       00001710
    2 YO=2.0*KOUNT*DELTAY                                               00001720
      XMAX=X(IHAVEP,ISWTCH)                                             00001730
    3 F=-TAN(MANGL)*XMAX-HO*SIN(XM*PI*XMAX)+YO                          00001740
      FP=-TAN(MANGL)-HO*XM*PI*COS(XM*PI*XMAX)                           00001750
      FAC=F/FP                                                          00001760
      XMAX=XMAX-FAC                                                     00001770
      IF(ABS(FAC).GT.0.00001)GO TO 3                                    00001780
      X(I,ISWTCH)=XMAX                                                  00001790
      R(I,ISWTCH)=HO*SIN(XM*PI*XMAX)+RO                                 00001800
      DELX(KOUNT)=XMAX-X(IHAVEP,ISWTCH)                                 00001810
      DELR(KOUNT)=R(IHAVEP,ISWTCH)-R(I,ISWTCH)                          00001820
      RETURN                                                            00001830
      END                                                               00001840
```

```
C                                                                       00000010
C            METHOD OF SINGULARITIES FOR TWO-BLADE CASCADE              00000020
C                 WITH SUBSONIC LEADING EDGE LOCUS                      00000030
C                                                                       00000040
      REAL M                                                            00000050
      DIMENSION PNR(2),PNI(2)                                           00000060
      COMMON/BLK1/PHIR,PHII,PHIXR,PHIXI,RF                              00000070
      READ(5,100)M,RF,GRDFN,Y,STGR                                      00000080
      WRITE(6,101)M,RF,GRDFN,Y,STGR                                     00000090
      WRITE(6,104)                                                      00000100
      BETA=SQRT(M**2-1.0)                                               00000110
      D=0.00001                                                         00000120
      N=0                                                               00000130
      DU=1.0/GRDFN                                                      00000140
      X=Y*BETA                                                          00000150
      STGR=STGR*0.01745                                                 00000160
    4 CONTINUE                                                          00000170
      DO 2 J=1,2                                                        00000180
      FOR=0.0                                                           00000190
      FOI=0.0                                                           00000200
      FOS=0.0                                                           00000210
      FOT=0.0                                                           00000220
      U=0.0                                                             00000230
      AA=BETA*Y                                                         00000240
      IF(J.EQ.1)GO TO 3                                                 00000250
      X=X-Y*TAN(STGR)                                                   00000260
      ARGC=0.0                                                          00000270
      ARGD=0.0                                                          00000280
      IF(X.EQ.0.0) GO TO 12                                             00000290
      IF(X.LT.0.0001)GO TO 7                                            00000300
      AA=0.0                                                            00000310
    3 CONTINUE                                                          00000320
      FAC=X-AA                                                          00000330
      XX=FAC/DU+1.0                                                     00000340
      LIMIT=XX                                                          00000350
      DXX=XX-LIMIT                                                      00000360
      IF(DXX.GE.0.5)LIMIT=LIMIT+1                                       00000370
      ARGA=(X-U)*RF*M**2/BETA**2                                        00000380
      ARGB=(RF*M/BETA**2)*SQRT(ABS((X-U)**2-AA**2))                     00000390
      ARGC=X*RF*M**2/BETA**2                                            00000400
      ARGD=RF*M*SQRT(ABS(X**2-AA**2))/BETA**2                           00000410
      IF(LIMIT.LE.1)GO TO 12                                            00000420
    8 CALL BESJ(ARGB,N,BJ,D,IER)                                        00000430
      A1=SIN(ARGA)*BJ*RF                                                00000440
      B1=SIN(ARGA)*BJ*RF                                                00000450
      C1=COS(ARGA)*BJ*RF                                                00000460
      D1=COS(ARGA)*BJ*RF                                                00000470
    9 DO 1 I=2,LIMIT                                                    00000480
```

```
      U=U+DU                                                                    00000490
      ARGA=(X-U)*RF*M**2/BETA**2                                                00000500
      ARGB=(RF*M/BETA**2)*SQRT(ABS((X-U)**2-AA**2))                             00000510
   10 CALL BESJ(ARGB,N,BJ,D,IER)                                                00000520
      A2=RF*(-RF*U*COS(ARGA)+SIN(ARGA))*BJ                                      00000530
      B2=BJ*SIN(ARGA)*RF                                                        00000540
      C2=RF*( RF*U*SIN(ARGA)+COS(ARGA))*BJ                                      00000550
      D2=BJ*COS(ARGA)*RF                                                        00000560
      FOR=FOR+(A1+A2)*DU/2.0                                                    00000570
      FOI=FOI+(B1+B2)*DU/2.0                                                    00000580
      FOS=FOS+(C1+C2)*DU/2.0                                                    00000590
      FOT=FOT+(D1+D2)*DU/2.0                                                    00000600
      A1=A2                                                                     00000610
      B1=B2                                                                     00000620
      C1=C2                                                                     00000630
      D1=D2                                                                     00000640
    1 CONTINUE                                                                  00000650
   12 CONTINUE                                                                  00000660
      CALL BESJ(ARGD,N,BJ,D,IER)                                                00000670
      PNR(J)=FOR+FOI+COS(ARGC)*BJ                                               00000680
      PNI(J)=FOS+FOT-SIN(ARGC)*BJ                                               00000690
      PNR(J)=PNR(J)/BETA                                                        00000700
      PNI(J)=PNI(J)/BETA                                                        00000710
      IF(J.EQ.2)GO TO 6                                                         00000720
      GO TO 2                                                                   00000730
    7 PNR(2)=0.0                                                                00000740
      PNI(2)=0.0                                                                00000750
      PHIR=0.0                                                                  00000760
      PHII=0.0                                                                  00000770
      PHIXR=0.0                                                                 00000780
      PHIXI=0.0                                                                 00000790
      VSR=0.0                                                                   00000800
      VSI=0.0                                                                   00000810
      X=X+Y*TAN(STGR)                                                           00000820
      GO TO 11                                                                  00000830
    6 CALL POTEN(X,VSR,VSI,M,BETA,STGR,Y,GRDFN)                                 00000840
   11 PN3R=PHIXR-RF*PHII                                                        00000850
      PN3I=PHIXI+RF*PHIR                                                        00000860
      PRESR=PNR(1)+PNR(2)-PN3R                                                  00000870
      PRESI=PNI(1)+PNI(2)-PN3I                                                  00000880
      WRITE(6,103)X,PNR(1),PNI(1),PNR(2),PNI(2),PHIR,PHII,PHIXR,PHIXI           00000890
      WRITE(6,102)PN3R,PN3I,VSR,VSI,PRESR,PRESI                                 00000900
    2 CONTINUE                                                                  00000910
      X=X+0.1                                                                   00000920
      IF(X.LE.Y*(TAN(STGR)+BETA))GO TO 4                                        00000930
  100 FORMAT(5F10.5)                                                            00000940
  101 FORMAT('1',10X,'MACH NUMBER=',F8.6,/,11X,'REDFRQ=',F7.2,/,11X,'GRI00000950
     1D FINENESS=',F7.2,/,11X,'Y=',F3.1,/,11X,'ZN(ZETA)=ZETA',/,11X,'BET00000960
```

```
   1A=',F5.2///)                                                                00000970
103 FORMAT('0',T5,F4.2,T13,F10.5,T27,F10.5,T41,F10.5,T55,F10.5,T69,F1000000980
   1.5,T83,F10.5,T97,F10.5,T111,F10.5)                                          00000990
102 FORMAT(' ',        T13,F10.5,T27,F10.5,T41,F10.5,T55,F10.5,T69,F1000001000
   1.5,T83,F10.5,T97,F10.5,T111,F10.5//)                                        00001010
104 FORMAT(' ',T7,'X',T16,'PNR(1)',T30,'PNI(1)',T44,'PNR(2)',T58,'PNI(00001020
   12)',T72,'PHIR',T86,'PHII',T100,'PHIXR',T114,'PHIXI',/,T16,'PNR(3)'00001030
   2,T30,'PNI(3)',T44,'VSR',T58,'VSI',T72,'PNR',T86,'PNI',/)                    00001040
    STOP                                                                        00001050
    END                                                                         00001060


    SUBROUTINE VSIGMA(Z,VSR,VSI,M,BETA,STGR,Y,GRDFN)                            00001070
    REAL M                                                                      00001080
    COMMON/BLK1/PHIR,PHII,PHIXR,PHIXI,RF                                        00001090
    D=0.00001                                                                   00001100
    N=1                                                                         00001110
    DU=1.0/GRDFN                                                                00001120
    X=Z+Y*TAN(STGR)                                                             00001130
  4 CONTINUE                                                                    00001140
    FOR=0.0                                                                     00001150
    FOI=0.0                                                                     00001160
    U=0.0                                                                       00001170
    ARGA=(X-U)*RF*M**2/BETA**2                                                  00001180
    ARGB=RF*M*SQRT(ABS((X-U)**2-(BETA*Y)**2))/BETA**2                           00001190
    ARGC=Y*RF*M**2/BETA**2                                                      00001200
    CALL BESJ(ARGB,N,BJ,D,IER)                                                  00001210
    A1= COS(ARGA)*BJ/SQRT(ABS((X-U)**2-(BETA)**2))                              00001220
    B1=-SIN(ARGA)*BJ/SQRT(ABS((X-U)**2-(BETA)**2))                              00001230
    XX=(X-BETA*Y)/DU+1.0                                                        00001240
    LIMIT=XX                                                                    00001250
    DXX=XX-LIMIT                                                                00001260
    IF(DXX.GE.0.5)LIMIT=LIMIT+1                                                 00001270
    DO 1 I=2,LIMIT                                                              00001280
    U=U+DU                                                                      00001290
    ARGA=(X-U)*RF*M**2/BETA**2                                                  00001300
    ARGB=RF*M*SQRT(ABS((X-U)**2-(BETA*Y)**2))/BETA**2                           00001310
    CALL BESJ(ARGB,N,BJ,D,IER)                                                  00001320
    A2=(RF*U*SIN(ARGA)+COS(ARGA))*BJ/SQRT(ABS((X-U)**2-(BETA*Y)**2))            00001330
    B2=(RF*U*COS(ARGA)-SIN(ARGA))*BJ/SQRT(ABS((X-U)**2-(BETA*Y)**2))            00001340
    FCR=FOR+(A1+A2)*DU/2.0                                                      00001350
    FCI=FOI+(B1+B2)*DU/2.0                                                      00001360
    A1=A2                                                                       00001370
    B1=B2                                                                       00001380
  1 CONTINUE                                                                    00001390
    FCR=FOR*RF*M*Y/BETA                                                         00001400
    FCI=FOI*RF*M*Y/BETA                                                         00001410
    S=-RF*(X-Y*BETA)*SIN(ARGC)-COS(ARGC)                                        00001420
```

```
      T=-RF*(X-Y*BETA)*COS(ARGC)+SIN(ARGC)                              00001430
      VSR=FOR+S                                                         00001440
      VSI=FOI+T                                                         00001450
      RETURN                                                            00001460
      END                                                               00001470

      SUBROUTINE POTEN(X,VSR,VSI,M,BETA,STGR,Y,GRDFN)                   00001480
      REAL M                                                            00001490
      COMMON/BLK1/PHIR,PHII,PHIXR,PHIXI,RF                              00001500
      U=0.0                                                             00001510
      DU=1.0/GRDFN                                                      00001520
      Z=0.0                                                             00001530
      ARGA=(X-U)*RF*M**2/BETA**2                                        00001540
      ARGB=(X-U)*RF*M/BETA**2                                           00001550
      A1=0.0                                                            00001560
      A2=0.0                                                            00001570
      B1=0.0                                                            00001580
      B2=0.0                                                            00001590
      B3=0.0                                                            00001600
      B4=0.0                                                            00001610
      XX=X/DU+1.0                                                       00001620
      LIMIT=XX                                                          00001630
      DXX=XX-LIMIT                                                      00001640
      IF(DXX.GE.0.5)LIMIT=LIMIT+1                                       00001650
      N=0                                                               00001660
      CALL BESJ(ARGB,N,BJ1,D,IER)                                       00001670
      N=1                                                               00001680
      CALL BESJ(ARGB,N,BJ2,D,IER)                                       00001690
      CALL VSIGMA(Z,VSR,VSI,M,BETA,STGR,Y,GRDFN)                        00001700
      IF(LIMIT.LE.1)GO TO 2                                             00001710
      PR1=(VSR*COS(ARGA)+VSI*SIN(ARGA))*BJ1                             00001720
      PI1=(VSI*COS(ARGA)-VSR*SIN(ARGA))*BJ1                             00001730
      PP1=(VSR*SIN(ARGA)-VSI*COS(ARGA))*BJ1                             00001740
      PP2=(VSR*COS(ARGA)+VSI*SIN(ARGA))*BJ2                             00001750
      PP3=(VSR*COS(ARGA)+VSI*SIN(ARGA))*BJ1                             00001760
      PP4=(-VSR*SIN(ARGA)+VSI*COS(ARGA))*BJ2                            00001770
      DO 1 I=2,LIMIT                                                    00001780
      U=U+DU                                                            00001790
      Z=U                                                               00001800
      ARGA=(X-U)*RF*M**2/BETA**2                                        00001810
      ARGB=(X-U)*RF*M/BETA**2                                           00001820
      N=0                                                               00001830
      CALL BESJ(ARGB,N,BJ1,D,IER)                                       00001840
      N=1                                                               00001850
      CALL BESJ(ARGB,N,BJ2,D,IER)                                       00001860
      CALL VSIGMA(Z,VSR,VSI,M,BETA,STGR,Y,GRDFN)                        00001870
      PR2=(VSR*COS(ARGA)+VSI*SIN(ARGA))*BJ1                             00001880
```

```
      PI2=(VSI*COS(ARGA)-VSR*SIN(ARGA))*BJ1                                    00001890
      PP5=(VSR*SIN(ARGA)-VSI*COS(ARGA))*BJ1                                    00001900
      PP6=(VSR*COS(ARGA)+VSI*SIN(ARGA))*BJ2                                    00001910
      PP7=(VSR*COS(ARGA)+VSI*SIN(ARGA))*BJ1                                    00001920
      PP8=(-VSR*SIN(ARGA)+VSI*COS(ARGA))*BJ2                                   00001930
      A1=A1+(PR1+PR2)*DU/2.0                                                   00001940
      A2=A2+(PI1+PI2)*DU/2.0                                                   00001950
      B1=B1+(PP1+PP5)*DU/2.0                                                   00001960
      B2=B2+(PP2+PP6)*DU/2.0                                                   00001970
      B3=B3+(PP3+PP7)*DU/2.0                                                   00001980
      B4=B4+(PP4+PP8)*DU/2.0                                                   00001990
      PR1=PR2                                                                  00002000
      PI1=PI2                                                                  00002010
      PP1=PP5                                                                  00002020
      PP2=PP6                                                                  00002030
      PP3=PP7                                                                  0C002040
      PP4=PP8                                                                  00002050
    1 CONTINUE                                                                 00002060
    2 PHIR=A1*(-1.0/BETA)                                                      00002070
      PHII=A2*(-1.0/BETA)                                                      00002080
      PHIXR=(M*B1+B2)*RF*M/(BETA**1.5)-(1.0/BETA)*VSR                          00002090
      PHIXI=(M*B3+B4)*RF*M/(BETA**1.5)-(1.0/BETA)*VSI                          00002100
      X=X+Y*TAN(STGR)                                                          00002110
      RETURN                                                                   00002120
      END                                                                      00002130
```

LIST OF REFERENCES

1. Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., Aeroelasticity, Addison-Wesley, 1955.

2. Dowell, E. H., and Widnall, S. E., Generalized Aerodynamic Forces on an Oscillating Cylinderical Shell, MIT, 1966.

3. Teipel, I., "Ein Charakteristikenverfahren zur Berchnung der verallgemeinerten ebenen Flatterluftkrafte," Zeitschrift fur Flugwissenschaften, v. 10, p. 374-379, October, 1962.

4. Leonard, R. W., and Hedgepeth, J. M., "On Panel Flutter and Divergence of Infinitely Long Unstiffened and Ring-Stiffened Thin-Walled Circular Cylinders," NACA Report 1302, 1957.

5. Matsuzaki, Y. and Kobayashi, S., "Unsteady Supersonic Aerodynamic Forces on an Oscillating Circular Cylindrical Shell," AIAA Journal 9, No. 12, p. 2358-2362, December 1971.

6. Dowell, E. H., Panel Flutter: A Review of the Aeroelastic Stability of Plates and Shells, AIAA Journal, Vol. 8, No. 3, March 1970.

7. Platzer, M. F., Brix, C. W., Jr. and Webster, K. A., "A Linearized Characteristics Method for Supersonic Flow Past Vibrating Shells," to be published in AIAA Journal, submitted March 1973.

8. Snyder, L. E., "Supersonic Torsional Flutter in Cascades," Final Technical Report PWA-4701, 2 April 1973, Pratt and Whitney Aircraft, East Hartford, Conn.

9. Chalkley, H. G., A Study of Supersonic Cascade Flutter, Aeronautical Engineer's Thesis, Naval Postgraduate School, March, 1972.

10. Webster, K. A., A Study of Supersonic Flow Past Vibrating Panels and Shells, Master's Thesis, Naval Postgraduate School, March, 1972.

11. Sauer, R., and Heinz, C., Unpublished report, see "Gasdynamics" by K. Oswatitsch, Academic Press, 1956, New York and London.

12. Garrick, I. E., and Rubinow, S. J., Flutter and Oscillating Air-Force Calculations for an Airfoil in Two-Dimensional Supersonic Flow, NACA Report 846, 1946.

13. Detroit Diesel Allison Report, "Aeroelasticity in Turbomachines," Proceedings of a workshop held at Detroit Diesel Allison, June 1-2, 1972. Project SQUID, Office of Naval Research.

14. Anderson, W. J., Oscillatory Pressure in an Idealized Boundary Layer with an Application to the Panel Flutter of Cylindrical Shells, AIAA Journal, Vol. 4, No. 5, May 1966.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Professor R. W. Bell, Code 57Be Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1
4.	Assoc. Professor M. F. Platzer, Code 57Pl Department of Aeronautics Naval Postgraduate School Monterey, California 93940	15
5.	LT C. W. Brix, Jr., USN GM DIV, Weapons Department USS Enterprise (CVAN-65) FPO San Francisco, California 96601	1
6.	Dr. H. J. Mueller (AIR - 310) Naval Air Systems Command Washington, D. C. 20360	1
7.	Professor M. H. Vavra, Code 57Va Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

ORIGINATING ACTIVITY (Corporate author)

Naval Postgraduate School
Monterey, California 93940

2a. REPORT SECURITY CLASSIFICATION

UNCLASSIFIED

2b. GROUP

REPORT TITLE

A Study of Supersonic Flow Past Vibrating Shells and Cascades

DESCRIPTIVE NOTES (Type of report and, inclusive dates)

Aeronautical Engineer's Thesis; June 1973

AUTHOR(S) (First name, middle initial, last name)

Christian W. Brix, Jr.

REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1973	133	14

CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
PROJECT NO.	
	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Postgraduate School Monterey, California 93940

ABSTRACT

Supersonic flow past oscillating cylindrical shells and oscillating flat-plate cascades is analyzed distributions and generalized aerodynamic forces are computed for arbitrary radius-to-length ratios. Axial and circumferential wave numbers, Mach number, and reduced frequencies. Excellent agreement is obtained with a previous solution of the complete unsteady linearized potential equation by Dowell and Widnall using Laplace transform techniques. Also, pressure distributions are computed for flat-plate cascades with subsonic leading edge locus for arbitrary solidity, stagger angle, frequency, and interblade phase angle. For comparison, a two-blade solution is developed using the method of singularities. Good agreement is obtained for this special case.

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Method of Characteristics						
Supersonic						
Cascade						
Cylindrical Shell						

Thesis
B80915 Brix
c.1
144296
A study of supersonic flow past vibrating shells and cascades.